

MasterSCADA™

Руководство пользователя

Версия 3.X



2017

Краткое описание руководства пользователя

Данное руководство предназначено для освоения программного обеспечения MasterSCADA и MasterPLC. Оно рассчитано как для начального освоения системы, так и для продвинутых пользователей.

В этом учебном пособии описаны основные принципы и методы, применяемые при разработке проектов в MasterSCADA, а также инструкции по установке исполнительной системы MasterPLC в контроллеры.

Адрес: 123298, г. Москва,
ул. Маршала Бирюзова, д. 1, корп. 3
Телефон: +7 (495) 989-22-49
Факс: +7 (499) 943-02-14 Email:
scada@insat.ru
Сайт: www.insat.ru

Содержание

Основные понятия	8
Интегрированная среда разработки MasterSCADA	9
Пользовательский интерфейс	9
Методика разработки системы	21
Описание технической структуры.....	21
Описание технологического объекта	23
Тиражирование элементов проектов и использование библиотек	26
Объектная идеология	27
Создание связей объекта с OPC переменными	28
Определение периода опроса	28
Работа с проектом	31
Элементы дерева системы	32
Элементы дерева объектов	45
Страницы свойств элементов.....	55
Архивы	109
Сообщения	113
Документы объектов.....	114
Управление ресурсами.....	118
Автоматическая установка периодического опроса	120
Назначение горячих клавиш	121
Подсчет количества связей.....	122
Работа в режиме исполнения	124
Пользовательский интерфейс в режиме исполнения	125
Страницы свойств в Режиме исполнения.....	128
Порядок вычислений и передачи данных.....	133
Восстановление состояния ФБ при рестарте.....	138
Резервирование компьютеров.....	138
Изменение списка операторов в режиме исполнения.....	140
Исполнительная система контроллера	141
Ограничение на использование ФБ и формул в контроллере	141
Подготовка контроллера к работе с MasterSCADA	142
Удаление файлов в контроллере	145
Автоматическое обновление исполнительной системы контроллеров.....	146
Архивирование в контроллере	146
Архитектура системы.....	147
Загрузка конфигураций.....	147
Резервирование контроллеров.....	148
Ресурсы контроллера	149
Исполнительная система контроллера	150
Соединение через GSM модем	151
Сообщения при загрузке	152

Типы задач исполнительной системы контроллера	152
Удаление конфигураций	153
Сетевые настройки MasterSCADA	154
Генератор отчетов	157
Выражения.....	157
Отображение текстовой информации.....	162
Отображение графической информации.....	172
Оформление компонентов.....	174
Создание отчета	177
Мастер отчетов	191
Секции отчета	193
Контейнеры.....	218
Использование авторазмера	221
Расчет итогов	227
Диаграммы.....	232
Навигация	239
Формирование отчетов по изменению значений	243
Свойства переменных источника данных	244
Модуль графиков и журналов сообщений.....	245
Методические указания по работе с трендом.....	245
Методические указания по работе с журналом	263
Справочная часть	270
Редактор схем функциональных блоков	307
Пользовательский интерфейс	307
Разработка схемы функциональных блоков	312
Палитра функциональных блоков.....	314
Обработка сигналов	314
Работа со временем	338
Исполнительные механизмы	342
Датчики	354
Аппараты	359
Вычисления	360
Генераторы значений.....	372
Управление.....	374
Функциональные блоки для работы с архивом.....	394
Служебные	404
Нечеткая логика	431
Другие	432
Отраслевая библиотека Электроэнергетика.....	470
Отраслевая библиотека Вентиляция.....	472
Общие свойства и настройки.....	519

Отображение недостоверных данных	522
Реакция на нажатие кнопок.....	523
Элементы мнемосхемы.....	525
Свойства объектов.....	525
Объёмные объекты.....	528
Предопределенные материалы	531
Объекты индикации и управления	533
Графические примитивы	544
Элементы диалога	550
Мультимедиа	557
Рисунки	557
Мультфильмы	557
Встроенный редактор мультфильмов.....	559
Настройка прозрачности	562
web-браузер.....	563
Метрологическая поверка	564
Порядок действий при настройке и проведении метрологической поверки	564
Общие сведения	565
Дополнительные сведения	568
Рекомендации по созданию отчетов.....	570
Служба шаблонных объектов	573

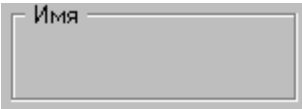






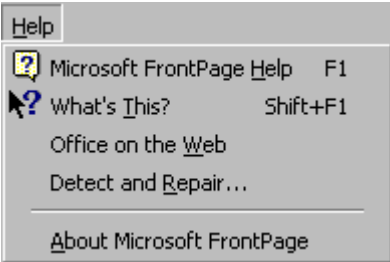
MasterSCADA™

Руководство пользователя

Версия 3.X

MasterSCADA™ — самый современный, инновационный мощный и удобный инструмент для быстрой и качественной разработки систем. Это программное обеспечение для систем управления, в котором воплощен двадцатилетний опыт разработчиков продуктов для автоматизации самых разных объектов.

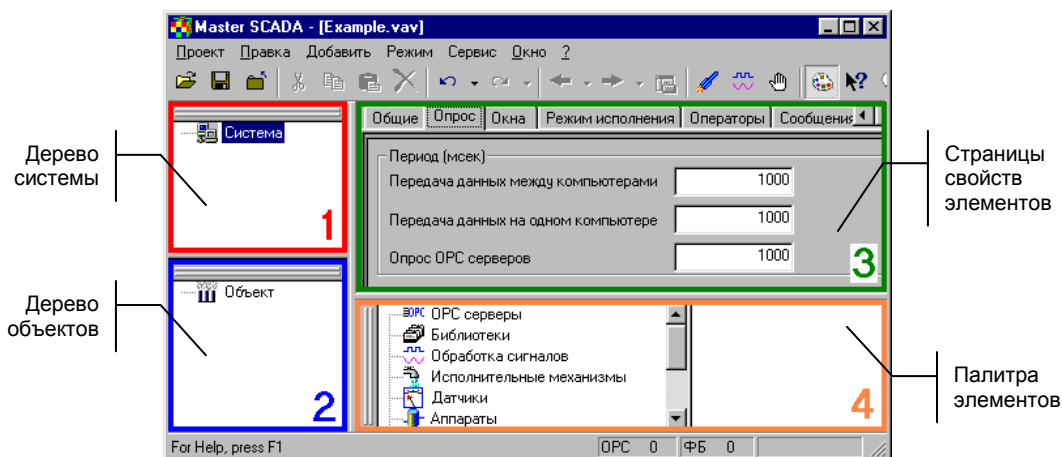
ОСНОВНЫЕ ПОНЯТИЯ

Термин	Значения
Группа	
Диалог	Окно с заголовком и элементами управления
Заголовок окна (в том числе - диалогового)	
Поле редактирования	
Кнопка	
Отмечаемая кнопка	
Радиокнопка (кнопка выбора)	
Элементы управления	Кнопки, поля редактирования, таблицы и. т. д. - все что можно встретить в окне диалога
Меню программы	 находится под заголовком окна программы
Командная группа	
Контекстное меню	Командная группа, раскрывающаяся при нажатии клавиши мыши (как правило правой) в том месте, где нажата клавиша. Зависит от того места, где находился указатель мыши в момент нажатия.

Интегрированная среда разработки MasterSCADA

Пользовательский интерфейс

Окно проекта состоит из четырех основных частей:



Размеры дерева системы, дерева объектов и палитры можно изменять и они могут располагаться в любом месте экрана. Страницы свойств занимают все оставшееся пространство.

Меню программы MasterSCADA

Основное меню программы состоит из следующих групп:

- Проект;
- Правка;
- Добавить;
- Режим;
- Сервис;
- Окно.

При открытии какой-либо документа для редактирования, меню изменяется, к нему добавляются группы из меню программы для редактирования документа и удаляются группы с командами, не имеющими смысла при редактировании документа.

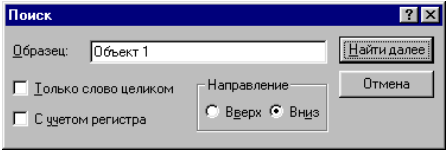
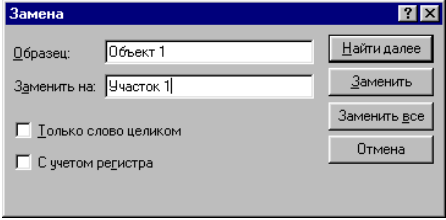
Меню Проект

Команда	Описание
Создать	Создание проекта.
Открыть	Открытие проекта.
Сохранить	Сохранить изменения в открытом документе.
Сохранить...	Позволяет переименовать проект и изменить пароль. Опция Оптимизировать структуру файла позволяет уменьшить размер файла проекта. Опция Пересохранить окна - все мнемосхемы проекта открываются в скрытом режиме и пересохраняются. Опция Создать конфигурационные файлы для контроллеров позволяет сохранить на диске файлы, которые загружаются в контроллер при запуске режима Real Time. Файлы сохраняются в подпапке проекта имя_компьютера/имя_контроллера : <i>config.bin</i> - двоичный файл, который грузится в контроллер; <i>config.xml</i> - xml файл для анализа конфигурации. Опция Кодировать имена папок идентификаторами позволяет зашифровать имена папок в проекте случайными идентификаторами, тем самым позволяя скрыть структуру проекта. Опция Запрет дальнейшего редактирования проекта сохраняет проект со случайным паролем, после этого он будет запускаться только в режим исполнения, но будет невозможно перейти в режим разработки. Удобно использовать для защиты вашего проекта после его передачи. Важно понимать что перед тем как использовать эту опцию необходимо сохранить копию проекта.

Команда	Описание
Сохранить как	Создать копию проекта (она может иметь другое имя). Все архивы (данных, сообщений, документов) при этом теряются. Сохранить как... отличается от экспорта следующим: Это действие можно выполнять на несохраненном проекте. Можно задать другое имя нового проекта. После выполнения действия в редакторе продолжается работа над копией проекта (после экспорта остается открытым оригинальный проект).
Сохранить все	Сохранить изменения в проекте.
Заккрыть	Заккрыть текущий проект.
Экспортировать	Экспорт проекта.
Импортировать	Импорт проекта.
Документировать	Создает словесное описание проекта в формате HTM или HTML, предварительно предложив задать имя и месторасположение файла.
Отчет об ошибках	Позволяет создать отчет об ошибках для обращения в службу поддержки.
Выход	Стандартное закрытие программы.

Меню Правка

Команды этой группы относятся к элементу, который в настоящий момент выделен в дереве. При этом команды, не имеющие смысла для элемента не доступны.

Команда	Описание
Отменить Вернуть	Позволяют соответственно отменить и восстановить отмененное ранее действие. Для отмены нескольких действий обратитесь к команде несколько раз, но удобнее воспользоваться соответствующим значком панели инструментов
Вырезать Копировать Вставить Удалить Дублировать Найти	Работают аналогично соответствующим командам из контекстного меню. Служит для поиска по имени нужного элемента в дереве. При использовании этой команды выводится такой стандартный диалог:
	
	Введите имя искомого элемента и нажмите Найти далее . Найденный элемент будет выделен в дереве (при условии, что он существует).
Заменить	Служит для замены имени элемента на другое. При использовании этой команды выводится такой стандартный диалог:
	
	При этом переименовать данной командой можно лишь названия, доступные для переименования.
Поместить в библиотеку	В палитре объектов существует специальная категория Библиотеки . Помещенные в нее объекты можно использовать многократно. Выделите объект, который необходимо сохранить для дальнейшего использования и нажмите Поместить в библиотеку . Объект будет скопирован и размещен в Библиотеке.

Меню *Добавить*

Добавление новых элементов в деревья можно осуществить через эту командную группу или через соответствующее контекстное меню. При выборе какой-либо из команд, соответствующий элемент будет добавлен в выделенный элемент в дереве, при условии, что вставку возможно произвести. Команды, невозможные для выполнения, при этом недоступны.

Меню *Режим*

Команда	Описание
Стоп	Включить режим разработки (Design mode).
Пуск	Включить режим исполнения (Runtime).
Имитация	Включить режим имитации.
Отладка	Включить режим отладки (Debug mode).

Меню *Сервис*

Настройка параметров среды MasterSCADA. Диалоговое окно настройки свойств среды вызывается по команде главного меню программы Сервис - Настройка

Элементы со страницы **общие** описаны ниже:

Элемент	Описание
Папка для хранения библиотек и настроек	В этой папке MasterSCADA хранит: [Library] - папка с библиотеками объектов; [Multimedia] - папка с библиотеками мультимедиа; [Projects] - папка с проектами, задаваемая по умолчанию; [Error Reports] - отчеты об ошибках MasterSCADA, создаваемые по команде пользователя; [Pictures] - копии рисунков; [Symbols] - библиотеки графических элементов MasterGraph; [Sys Log] - текущие отчеты о ошибках MasterSCADA; [Temp] - папка с временными файлами; TreeFileStorage - файл с настройками палитры MasterGraph; visavis.stg - файл с настройками палитры MasterSCADA.
Программа для просмотра журнала	Разрабатывается, пока не используется.
Навигация по дереву	Настраивается какие закладки открывать при выделении в дереве другого элемента.
Запрет поиска внешних связей при вставке	Установка флажка отменяет восстановление внешних связей при копировании объектов.
При сохранении мнемосхемы создавать изображение для кнопки объекта	Создание отдельных изображений кнопок объектов при закрытии мнемосхемы данного объекта.
Режим обнаружения ошибок	Рекомендуется включать этот режим при отладке проектов. Это незначительно замедляет работу программы, но делает возможным создание отчетов об ошибках.

Элементы со страницы **режимов исполнения** описаны ниже:

Элемент	Описание
Максимальное количество рабочих потоков	Пересчет каждого цикла объекта производится в отдельном потоке. Если в проекте много циклов вычислений, имеет смысл увеличить максимальное число рабочих потоков.
Буфер данных для просмотра ___ минут	Отсюда берется значение по умолчанию для страницы Данные переменных.
Максимальный размер буфера данных для просмотра	Настраивается максимальное количество данных для просмотра (от 1 до 10000000).
Формирование полного журнала режима исполнения	При установленном флаге формируется файл Runtime.log в который записываются все данные режима исполнения (нужен для технической поддержки).
Максимальное количество действий в очереди	Действия, требующие выполнения не запускаются сразу - сперва они ставятся в очередь, из которой впоследствии выбираются MasterSCADA и запускаются. Настройка позволяет указать максимальный размер очереди действий.

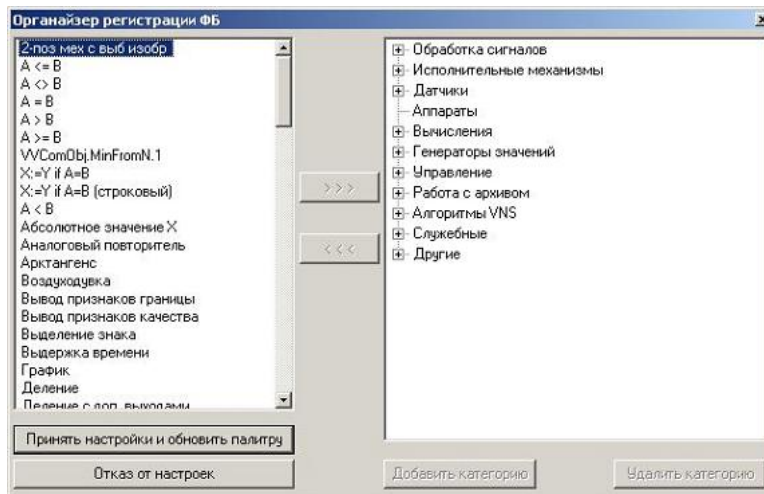
Элемент	Описание
Таймаут выполнения цикла объекта	Настраивается время выполнения функциональных блоков и следующих операций: Передача данных в документ (мнемосхема, тренд) Передача данных по сети к другим компьютерам. Если по истечении данного промежутка времени операция не выполнена, то она принудительно завершается (при этом выдается сообщение), и начинается выполнение следующей операции.
Запуск мнемосхем в отдельном процессе	Позволяет определить, в каком процессе в режиме исполнения следует запускать мнемосхемы - в процессе MasterSCADA или в отдельном процессе. При запуске мнемосхем в процессе MasterSCADA значительно увеличивается производительность графической подсистемы (за счет отсутствия передачи данных между границами процессов), но, естественно, несколько снижается надежность.
Период сброса архивов на диск	Для увеличения производительности системы в режиме исполнения, новые данные, поступающие в архивы, пишутся сначала в специальные буферы, находящиеся в оперативной памяти, и только потом в файлы на диске. Настройка позволяет указать, через какой промежуток времени необходимо сбрасывать эти буферы на жесткий диск компьютера.
Формировать признаки качества выходов ФБ по признакам качества входов	Флаг устанавливается для того чтобы выходы функциональных блоков или результат формулы (для переменной Расчет) наследовали от входов (или аргументов формулы) значения их признаков качества (недоверности) в соответствии со стандартом OPC. Примечание: на текущий момент реализовано только для следующих ФБ: Расчет, Упаковка 32 дискретных значений, Распаковка аналогового 32 бит. значения.

Обновление палитры ФБ и графических библиотек. Процедуры обновления палитры ФБ и графических библиотек запускаются по командам главного меню программы:





- Сервис - Обновить - Палитру ФБ
Необходимо использовать эту команду при изменении каких-либо настроек в палитре, например, после регистрации новой библиотеки функциональных блоков, для того, чтобы палитра и ее внешний вид изменились в соответствии с новыми параметрами (перезапуска программы не всегда бывает достаточно). Так же эта команда может понадобиться при переносе библиотечных объектов, созданных на другом компьютере.
При первом запуске программы палитра обновляется автоматически. Также возможно автоматическое обновление палитры при последующих запусках, если производятся какие-нибудь изменения.
- Сервис - Обновить - Графические библиотеки
Обновляет палитру элементов MasterGraph

Организатор регистрации функциональных блоков. Для запуска выберите пункт **Организатор регистрации функциональных блоков** в меню **Сервис**. Появится окно организатора.

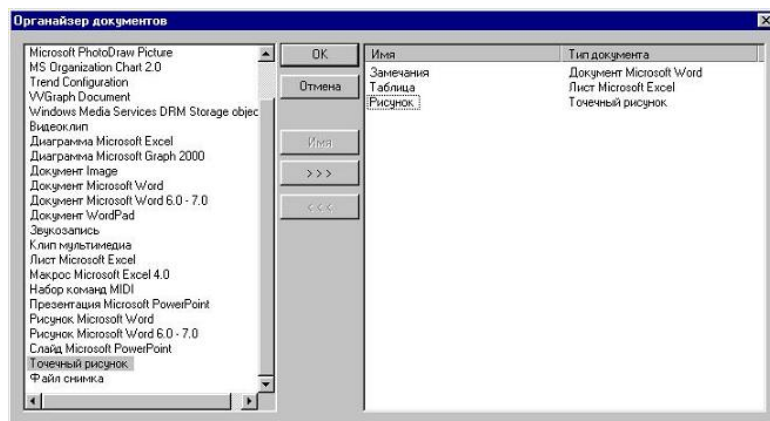
Организатор регистрации функциональных блоков обеспечивает для пользователя возможность в случае необходимости настроить палитру элементов согласно собственным предпочтениям. В правой части картинки выше можно наблюдать дерево функциональных блоков, вид которого можно настраивать.





В процессе настройки категорий возможно вставлять и удалять только категории второго уровня при помощи соответствующих кнопок внизу справа. Добавлять и удалять блоки можно в категориях и первого и второго уровня. Для вставки – необходимо выделить блок слева

и категорию справа и нажать . Для удаления - выделить блок справа и нажать . Для сохранения изменений необходимо нажать кнопку , если же не требуется использовать внесенные изменения, нажмите  и органайзер будет закрыт без сохранения новой конфигурации.

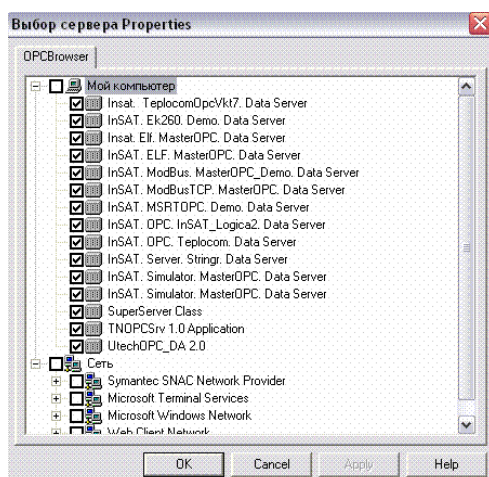
Органайзер документов. Диалоговое окно органайзера документов вызывается по команде главного меню **Сервис - Органайзер - Документов**.



В левой части органайзера перечислены все зарегистрированные в системе типы документов. В правой части - список документов, которые должны быть доступны. Для помещения типа документа в список выделите его в левой части и нажмите кнопку , для удаления выделенного в правой части типа документа нажмите .

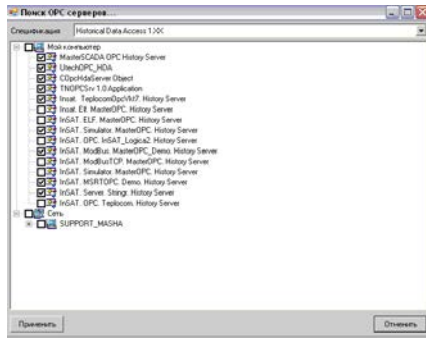
Для каждого типа документов можно задать свое название, под которым он будет храниться в объекте. Для этого выделите строку справа с именем, которое хотите поменять и нажмите кнопку **Имя**, после чего введите свое название. Кнопки **OK** и **Отмена** предназначены, как обычно для выхода из органайзера с сохранением изменений и без сохранения соответственно. На закладке **Другие** свойств **Объекта** при нажатии кнопки **Добавить** появится окно со списком тех типов документов, которые ранее были сделаны доступными.

Органайзер OPC DA серверов. Диалоговое окно браузера OPC серверов вызывается по команде главного меню **Сервис - Органайзер - OPC DA Серверов**, или через контекстное меню **Компьютера**. В результате на экране появится органайзер как на картинке ниже.



В окне органайзера отображено дерево OPC DA серверов, обнаруженных на данном компьютере и в ближайшем сетевом окружении. Отметьте флажками OPC DA серверы, которые будут использоваться в проекте и впоследствии они будут отображены в контекстном меню **Компьютера**, раскрывающейся команде **OPC серверы** в меню **Добавить**, а также в категории **OPC серверы** в палитре элементов.

Органайзер OPC HDA серверов. Диалоговое окно браузера OPC серверов вызывается по команде главного меню **Сервис - Органайзер - OPC HDA Серверов**, или через контекстное меню **Компьютера**. В результате на экране появится органайзер как на картинке ниже.



В окне органайзера отображено дерево OPC HDA серверов, обнаруженных на данном компьютере и в ближайшем сетевом окружении. Отметьте флажками OPC HDA серверы, которые будут использоваться в проекте и впоследствии они будут отображены в контекстном меню **Компьютера**, раскрывающейся команде "OPC серверы" в меню "Добавить", а также в категории "OPC серверы" в палитре элементов.

Меню Окно

В меню **Окно** можно выбрать, какой из открытых проектов отображать на экране в данный момент. Одновременно может быть на экране только один из открытых проектов. В режиме RunTime - лишь один проект может быть открыт.

Панель инструментов программы MasterSCADA

Кнопка	Описание
	Создать новый проект;
	Открыть существующий проект;
	Сохранить текущий проект;
	Закрыть текущий проект;
	Перейти к редактированию предыдущих документов;
	Перейти к редактированию следующего документа;
	Показать свойства проекта;
	Вырезать выделенные элементы в буфер;
	Копировать выделенные элементы в буфер;
	Вставить содержимое буфера;
	Удалить выделенные элементы;
	Отмена одного (по умолчанию) или нескольких (в раскрывающемся списке) из последних выполненных действий;
	Восстановление одного или нескольких отмененных ранее действий;
	Запустить / остановить режим имитации;
	Запустить / остановить режим исполнения (RunTime);
	Запустить / остановить режим отладки;
	Показать / скрыть палитру элементов;
	Показать / скрыть все элементы дерева объектов
	Переходит в режим редактирования схемы функциональных блоков выделенного объекта
	Контекстная справка;
	Сведения о программе.

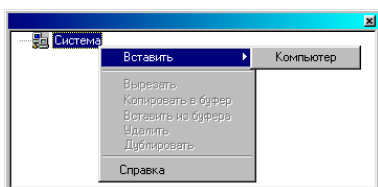
Контекстное меню

Контекстные меню открываются при нажатии правой клавиши мыши на каком-нибудь элементе в дереве системы или дереве объектов. При этом у различных элементов контекстные меню могут отличаться как набором команд, так и их доступностью.

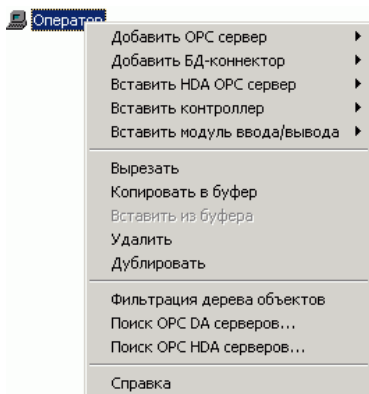
Можно выделить их следующие основные типы:

- Элементы дерева Системы:
 - Контекстное меню Системы;
 - Контекстное меню Компьютера;
 - Контекстное меню OPC сервера и группы OPC переменных.
- Элементы дерева объектов:
 - Контекстное меню главного Объекта;
 - Контекстное меню вложенного Объекта;
 - Контекстное меню Функционального блока в дереве.
- Элементы палитры:
 - Контекстное меню Функционального блока в палитре;
 - Контекстное меню Объекта в библиотеке.

Контекстное меню Системы позволяет выполнить единственное действие – перемещение **Компьютера** в **Систему**.



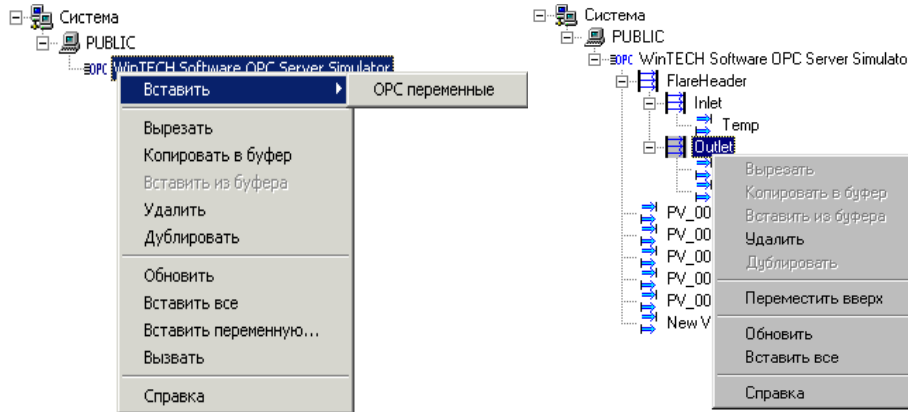
Контекстное меню Компьютера позволяет выполнить следующие действия:



Действие	Комментарий
Добавить OPC-сервер	Вставить OPC DA-сервер в Компьютер .
Добавить БД-коннектор	Вставить коннектор для обмена с базой данных в Компьютер .
Вставить HDA OPC- сервер	Вставляет OPC HDA - сервер в Компьютер .
Вставить контроллер	Вставляет контроллер в Компьютер .
Вставить модуль ввода/вывода	Вставляет модуль ввода/вывода в Компьютер .
Вырезать	Вставить в буфер обмена.
Копировать в буфер	Копировать Компьютер в буфер обмена.
Вставить из буфера	Вставить из буфера обмена (доступно, если в буфере находится ранее скопированный туда OPC-сервер).
Удалить	Удалить Компьютер из дерева Системы.
Дублировать	То же, что Копировать , но с возможностью задания числа копий
Переместить вверх	Перемещение Компьютера вверх по дереву системы относительно других компьютеров (отсутствует, если компьютер верхний).
Переместить вниз	Перемещение Компьютера вниз по дереву системы относительно других компьютеров (отсутствует, если компьютер нижний).

Действие	Комментарий
Фильтрация дерева объектов	Результатом выполнения этой операции будет выделение в дереве объектов жирным шрифтом имен тех элементов, которые используют данный компьютер.
Поиск OPC DA серверов	Поиск OPC DA серверов. Подробное описание дано в последующих разделах.
Поиск OPC HDA серверов	Поиск OPC HDA серверов. Подробное описание дано в последующих разделах.
Справка	Вывод контекстной справки.

Контекстное меню OPC серверов

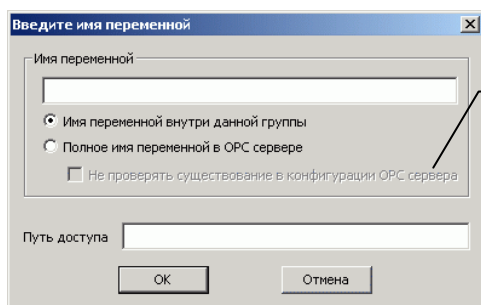


Некоторые команды контекстного меню, а именно:

- Вырезать;
- Копировать в буфер;
- Вставить из буфера;
- Удалить;
- Дублировать;
- Переместить вверх;
- Переместить вниз;
- Справка.

полностью идентичны подобным командам контекстного меню **Компьютера** описанным выше. Описание остальных команд указано в таблице ниже:

Команда	Комментарий
Вставить	Раскрывает пункт OPC переменные.
Обновить	Позволяет обновить список тегов при удалении или смене названия, а также при смене типов переменных и дополнительных свойств непосредственно в OPC-сервере.
Вставить все	Означает выполнение сразу двух действий. Сначала выполняется Обновить , после чего вставляются все созданные в программе OPC сервер переменные. Для группы OPC переменных эта команда равносильна тому, что в окне Выбор групп и переменных данной группе установили флажок и нажали ОК.
Вставить переменную	Открывает диалоговое окно, позволяющее вставить OPC переменную по имени.
Вызвать	Запуск пользовательского интерфейса программы OPC сервер.



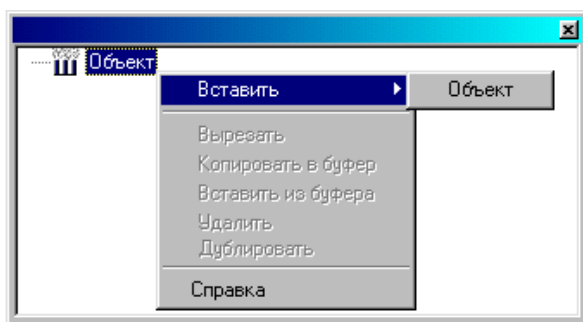
Диалоговое окно ввода переменной по имени

Если в диалоговом окне ввода переменной выбран переключатель **Имя переменной внутри данной группы**, то поле **Имя** должно содержать относительный путь к переменной от текущего элемента (для которого было вызвано контекстное меню). Если путь, заданный в поле **Имя** содержит точки, то в процессе добавления переменной также будут созданы все нужные группы.

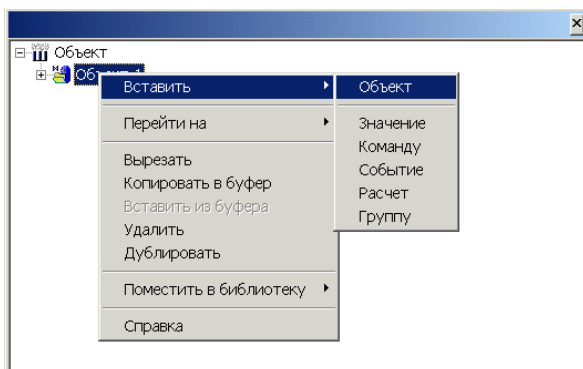
При выборе переключателя **Полное имя переменной в OPC сервере**, в поле **Имя** должен быть задан полный путь к переменной в OPC сервере. При вставке переменная добавляется в текущий элемент дерева системы (OPC сервер или группу), для которого было открыто контекстное меню. Переключатель **Не проверять существование в конфигурации OPC сервера** позволяет отключить проверку наличия заданной переменной в конфигурации OPC сервера.

Возможность вставки переменной, используя **Путь доступа** (также может именоваться AccessPath) зависит от конкретного OPC сервера. Чтобы узнать поддерживает ли OPC сервер эту функцию, обратитесь к его документации. Если **Путь доступа** задан, то MasterSCADA пытается добавить переменную с именем, введенным в поле **Имя**, согласно настроенному пути доступа. После вставки, в дереве системы она будет иметь имя в виде **[Путь доступа]ИмяПеременной**.

В **Контекстном меню главного Объекта** возможна только вставка подчиненного объекта. Вид контекстного меню предоставлен на картинке ниже.



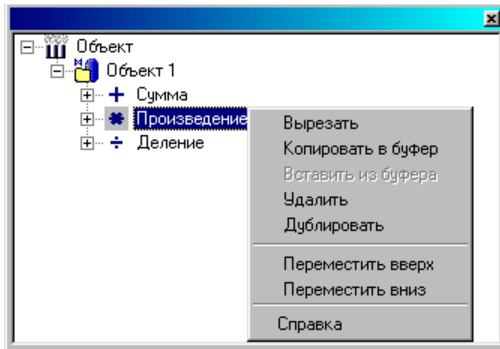
Контекстное меню вложенного объекта



Действия повторяющихся в других контекстных меню команд идентичны соответствующим командам контекстного меню Компьютера. При выборе пункта контекстного меню **Перейти на**, открывается меню следующего уровня, в котором перечислены все документы, входящие в состав объекта. В режиме исполнения в этот список добавляются журналы объекта, т.к. открытие журнала доступно только в режиме исполнения. После открытия какого-либо документа, как в режиме исполнения, так и в режиме разработки, объект, которому принадлежит данный документ выделяется в дереве жирным шрифтом.

Применение и описание команд **Поместить в библиотеку**, **Вставить Значение**, **Вставить Команду**, **Вставить Событие**, **Вставить Расчет**, **Вставить Группу** более подробно описано в последующих разделах.

Контекстное меню функционального блока в дереве



Команды **Вырезать**, **Копировать в буфер**, **Удалить** и **Дублировать** - идентичны соответствующим командам контекстного меню **Компьютера**. Команды **Переместить вверх** и **Переместить вниз** предназначены для управления положением блока в дереве. В ряде случаев является важным то обстоятельство, что пересчет блоков внутри **Объекта** (папки) производится программой сверху вниз.

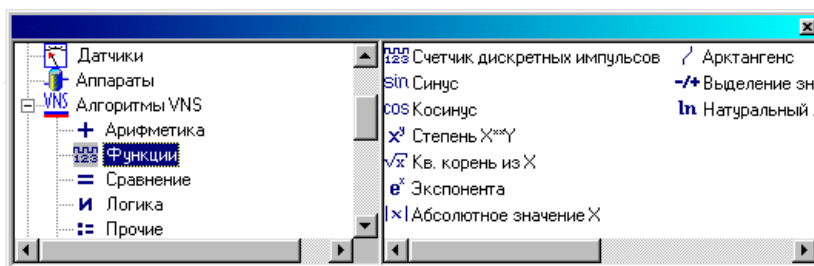
Контекстное меню функционального блока в палитре имеет команду **Зафиксировать**, которая позволяет *прицепить* его к указателю мыши до тех пор, пока не будет нажата правая клавиша мыши в области палитры. Это бывает удобно в случае, если необходимо разместить один и тот же блок в нескольких местах в дереве. Вызов **Справки** работает только для тех блоков, для которых разработчик создал контекстную справку.

Контекстное меню Объекта в библиотеке имеет команду **Зафиксировать** работающую по принципу команды из контекстного меню ФБ в палитре.

- ❖ **Важно!** После удаления объекта из библиотеки с помощью команды **Удалить** восстановление будет невозможно.

Палитра элементов


Палитра состоит из двух основных частей. Слева находится дерево категорий элементов палитры, справа - собственно элементы, которые используются при создании проекта. Палитра может *плавать* или быть *приклеенной* к краю экрана в зависимости от желания пользователя.



Элементы сгруппированы по категориям для удобства. Принадлежность элемента к той или иной категории может быть изменена с помощью органайзера регистрации функциональных блоков.

- В категории **ОПС серверы** находятся ОПС серверы, которые программа автоматически помещает сюда после использования команды **Поиск ОПС серверов**. Для использования ОПС сервера достаточно сделать на нем щелчок мышью, после чего перевести изображение в дерево системы, и после нажатия на каком-либо компьютере ОПС сервер будет в него помещен и готов к использованию.
- Категория **Обработка сигналов** содержит набор функциональных блоков, так или иначе связанных с обработкой сигналов. Часть их дублирует Алгоритмы VNS.
- В категории **Исполнительные механизмы** содержатся визуальные функциональные блоки, предназначенные для управления исполнительными механизмами.
- Категория **Датчики** содержит ФБ, предназначенные для контроля и отображения данных на мнемосхеме. Для контроля и управления аппаратами и исполнительными механизмами необходимо воспользоваться функциональными блоками из соответствующих категорий **Аппараты** или **Управление**.

- В категории **Генераторы значений** содержатся функциональные блоки предназначенные для формирования дискретных и аналоговых сигналов по различным законам (пила, пульсатор, шум и т.п.). Функции остальных функциональных блоков более подробно описаны в контекстной справке.

Для получения сведений о конкретном функциональном блоке воспользуйтесь контекстной справкой (кликните по иконке , затем по интересующему функциональному блоку). Дополнительные ФБ разработанные пользователем хранятся в отдельной категории **Другие**

Работа с элементами

Для того, чтобы вставить в дерево новый компьютер, OPC сервер, переменную, группу переменных или новый объект - используйте контекстное меню вызываемое правой кнопкой мыши, кнопку *Insert* или одну из команд меню **Добавить**.

Выбор нужного элемента (например, функционального блока) в палитре элементов осуществляется нажатием левой клавиши мыши на элементе, для сброса - нажмите правую клавишу. Выбранный элемент перемещается мышью в нужное место в дереве и вставляется нажатием левой клавиши. При этом левый верхний угол значка объекта в случае надобности можно использовать как указатель мыши, например для перемещения полосы прокрутки, то есть изображение элемента по существу является курсором.

Перемещение, копирование, дублирование и удаление элемента в дереве можно осуществить через контекстное меню объекта или элемента, всплывающее при нажатии правой клавиши мыши на нужном элементе, с помощью кнопок панели инструментов, или использовать горячие клавиши описанные ниже.

При дублировании объекта во всплывающем диалоге можно задать сколь угодно большое количество копий. Копии будут располагаться на том же уровне, что и исходный блок. Копирование и дублирование производится включая все вложенные объекты. Вставка из буфера осуществляется в выбранный элемент дерева, при этом, если выбранный элемент не предусматривает возможность вставки в себя других блоков, вставки не произойдет и на экран будет выведено соответствующее сообщение. Так же копировать элементы деревьев проекта (переменные объекта, функциональные блоки, группы объекта, группы выделенных элементов и т.д.) можно путем простого перетаскивания.

В дереве объекта можно использовать выделение группы элементов подобно тому, как это делается в программе *Explorer*. Если щелкнуть левой клавишей мыши на элементе, удерживая нажатой клавишу **Ctrl**, то элемент будет добавлен к уже выделенным, если же удерживать клавишу **Shift**, будут выделены все элементы от ранее выделенного до текущего, находящиеся на одном уровне вложенности.

Горячие клавиши

Сочетание клавиш	Действие
Ctrl + C	Копировать в буфер
Ctrl + F	Найти
Ctrl + H	Заменить
Ctrl + N	Создать новый проект
Ctrl + O	Открыть проект
Ctrl + P	Печать
Ctrl + S	Сохранить
Ctrl + V	Вставить из буфера
Ctrl + X	Вырезать
Ctrl + Y	Вернуть отмененное действие
Ctrl + Z; Alt + Back	Отменить последнее действие
F5	Перейти в режим исполнения
Shift + F5	Выйти из режима исполнения

Сочетание клавиш	Действие
F6	Перейти к следующему документу
Shift + F6	Перейти к предыдущему документу
F11	Переход в полноэкранный режим, и возврат из полноэкранного режима.
Shift + стрелки вверх и вниз	Перемещение
Del	Удалить

Строка состояния

Строка состояния представляет собой горизонтальную полосу в нижней части окна MasterSCADA и предназначена для отображения текущего состояния проекта



В основном теле **Строки** состояния пишется подсказка, относящаяся к содержимому открытого окна проекта. С правой стороны находится информация о количестве разного типа переменных проекта (ОПС-переменные, выходы функциональных блоков, переменные объектов). Так же, Строка состояния содержит кнопку вызова **Протокола работы** пакета MasterSCADA в режиме исполнения. Кнопка находится в крайнем правом углу.

Методика разработки системы

Разработка проекта в MasterSCADA состоит из двух основных частей:

- описания технической структуры системы;
- описание технологического объекта.

Каждая часть в свою очередь состоит из нескольких этапов (шагов). Выполнив шаг за шагом весь процесс создания проекта, пользователь получает готовую к работе систему управления. Порядок, в котором необходимо выполнять описания не имеет особого значения, более того, он может выполняться отдельными разработчиками. При настройке системы устанавливаются параметры, которые будут впоследствии унаследованы подчиненными объектами. Настраивая технологические объекты, можно отложить назначение объекту компьютера до последнего момента. При этом, определите характерные параметры объектов, оставив без изменений те, которые являются общими для системы. Как только объекту будет назначен компьютер, общие настройки будут автоматически унаследованы. Этот раздел справки содержит краткое описание разработки системы.

Описание технической структуры

Построение технической структуры распределенной системы выполняется за четыре шага:

- *Шаг первый.* Описание настроек связи с контроллерами и формирование групп опрашиваемых переменных;
- *Шаг второй.* Описание структуры создаваемой АСУТП;
- *Шаг третий.* Задание общих настроек системы;
- *Шаг четвертый.* Задание общих настроек отдельных компьютеров.

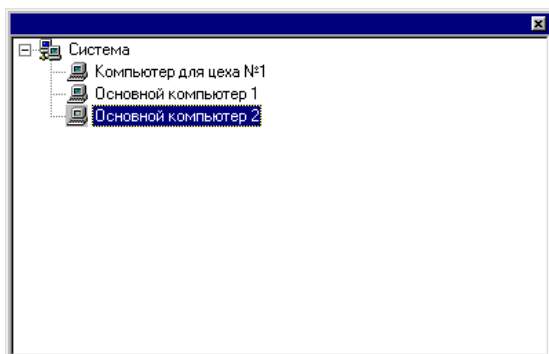
Описание настроек связи с контроллерами

Производится непосредственно в OPC серверах контроллеров. Так как OPC серверы имеют индивидуальные пользовательские интерфейсы, рекомендации по настройке переменных следует смотреть в документации конкретного OPC сервера. После выполнения необходимых настроек OPC переменных в OPC серверах можно переходить к следующему шагу - созданию структуры АСУТП.

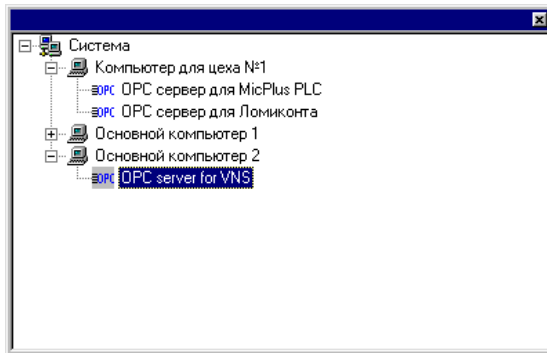
Описание структуры АСУ ТП

Настройка структуры АСУТП с помощью дерева системы выполняется следующим образом:

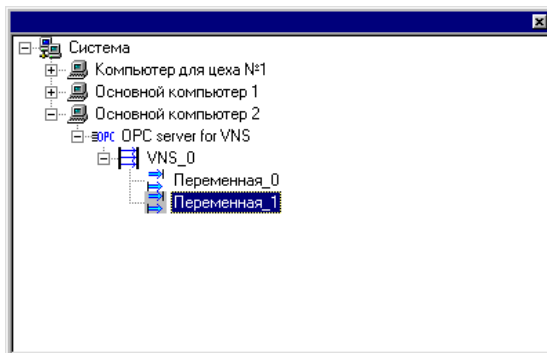
- В дерево *Системы* поместите необходимое число компьютеров (с помощью *контекстного меню* или меню *Добавить*). На закладке *Общие* у компьютера задайте каждому компьютеру имя, напишите комментарий (если нужно).



- В компьютеры поместите (тем же способом) *OPC серверы*. При первом запуске нужно запустить *Поиск OPC серверов* и в появившемся *OPC органайзере*.



- Добавьте в OPC серверы необходимые *OPC переменные*. Можно использовать команду *Вставить все*. Переменные немедленно будут видны в дереве системы и готовы к использованию



Задание общих настроек системы

Перемещаясь по закладкам свойств *Системы* необходимо задать общие настройки, которые затем наследуются всеми компьютерами, подчиненными элементами компьютеров и т. д., у которых не устанавливали собственных значений. Если же какой-либо параметр у подчиненного элемента изменяется *вручную*, то он больше не будет его наследовать. Таким образом, все настройки лучше производить *сверху вниз*. Для того, чтобы было удобнее отличать параметры, унаследованные от *родительского элемента*, они выделяются серым цветом.

- На закладке *Общие* можно только написать комментарий (это - просто текст), если он нужен;
- Закладка *Опрос* имеет два листа - *Опрос* и *Имитация*.

На листе *Опрос* устанавливаются параметры передачи данных. Установленные настройки можно будет изменить при последующих настройках подчиненных элементов в *индивидуальном* порядке. Поэтому значения лучше выбрать те, которые понадобятся большинству подчиненных элементов, чтобы потом вносить меньше изменений.

На листе *Имитация* устанавливаются параметры имитации переменных на входах если это необходимо.

Закладка *Окна* имеет таблицу, в которой устанавливаются типы и размеры окон документов. Размеры впоследствии можно будет изменить применительно к каждому компьютеру, поэтому установите наиболее часто употребляемые. Типы же окон устанавливаются для всей системы сразу и впоследствии возможно лишь выбирать для документа окно из существующих типов. А добавлять-удалять типы можно только в этом месте.

Закладка *Режим исполнения* имеет также два листа: лист *Настройки* содержит ряд параметров для наследования компьютерами, лист *Протокол работы*. В протоколе работы фиксируются ошибки в системе.

Закладка *Операторы* имеет три листа - *Права доступа*, *Смены* и *Должности*. На них производятся общие для системы настройки. На листе *Должности* добавьте в таблицу какие считаете нужными должности операторов. Впоследствии, при настройке компьютеров, необходимо выбирать должность для конкретного оператора из настроенного здесь списка. Количество и время смен, настроенные на листе *Смены*, можно будет менять при настройке компьютеров.

Закладка *Сообщения* соответственно служит для настройки сообщений. На закладке *Архив* установите параметры, характерные для большинства компьютеров.

Закладка *Шкалы* позволяет вводить собственные интервалы и единицы измерения, если не достаточно предопределенных. Интервалы можно использовать следующим образом: на закладке *Общие* у переменных выберите настроенный интервал и переключитесь на закладку контроль значений. Как видно, все поля этой закладки уже заполнены автоматически значениями, установленными для выбранного интервала.

Задание настроек компьютера

Задаются только те настройки каждого компьютера, которые на данной машине по каким-либо причинам отличаются от унаследованных настроек системы, например: размер экрана, объем дискового пространства, отводимый под архив и прочее в том же роде. Переходя по страницам свойств *Компьютеров* внесите необходимые изменения.

- Закладка *Общие* - для задания имени и комментария;
- Закладка *Опрос* - унаследовала параметры, установленные для системы. При необходимости их можно изменить, если на данном компьютере используются отличные от общих значения;
- На закладке *Окна* установите размеры окон, если не подходят унаследованные от системы. Впоследствии документы, созданные для объектов, которым будет назначен данный компьютер, будут иметь именно эти размеры;
- Закладка *Режим исполнения* - аналогичная листу *Настройки* у одноименной закладки свойств *Системы*, откуда она наследует данные. Здесь можно выбрать стартовую заставку - изображение, которое появляется на экране во время перехода в режим исполнения, назначить вид после старта - начинать ли работу с запуска стартовой мнемосхемы (для создания и редактирования стартовой мнемосхемы есть кнопка на закладке *Окна*), или же оставить на экране страницы свойств проекта;
- Закладка *Сообщения* - похожа на такую же закладку у *Системы*, откуда она наследует данные (отличается только наличием кнопки *Восстановить умолчания*);
- Закладка *Архив* - установите размеры файлов архивов для данного компьютера;
- На закладке *Операторы* под каждой должностью (должности настраиваются для системы сразу) укажите имена операторов, которые будут работать за данным компьютером. Для каждого оператора потом задается самым главным администратором пароль;
- На закладке *Смены* внесите изменения, если понадобится, в порядок смен. Для каждой смены назначьте операторов, установив флажки напротив соответствующего имени.

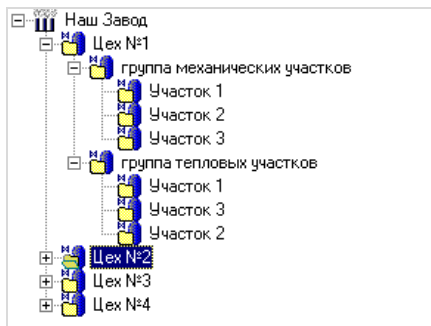
Описание технологического объекта

Также, как описание технической структуры АСУ ТП, описание технологического объекта выполняется за четыре шага:

- Шаг первый. Создание дерева объекта;
- Шаг второй. Размещение технологических агрегатов;
- Шаг третий. Создание списка переменных элемента дерева объекта;
- Шаг четвертый. Настройка Объектов.

Создание дерева объектов

Разработайте *Дерево Объектов* в соответствии с реальной схемой вашего технологического процесса, например, завод состоит их цехов, цеха из участков и т. д. При этом удобно пользоваться командами *контекстного меню*.



На закладке *Общие* каждого объекта задайте ему имя (можно и в дереве) и комментарий (если нужно). Можно сразу произвести и некоторые другие настройки свойств объектов, но удобнее сделать это позднее (на четвертом шаге).

Размещение технологических агрегатов в дереве объекта

Участки состоят из технологических агрегатов. Если агрегаты типовые, то просто перетащите их из *библиотеки*, которая всегда находится под рукой в составе *палитры элементов*. Если же встретился нетиповой агрегат, то надо создать этот агрегат в дереве объекта, настроить свойства, включить в его состав библиотечные насосы, задвижки, клапаны, а затем поместить его в библиотеку, чтобы затем использовать как типовой элемент системы. Не следует опасаться перегрузить библиотеку, ее размер ни на что не влияет. А иметь под рукой множество типовых элементов всегда удобно, даже если кое-что придется подправить. Нетиповой элемент необходимо полностью настроить. Например, если создается какой-то агрегат, необходимо произвести следующие действия:

- Сначала поместите в нужное место отдельный объект для агрегата и задайте ему соответствующее имя на закладке *Общие*, если это не сделано при разработке дерева объектов;
- Поведение элемента определяется схемой из функциональных блоков, которые вставляются в дерево объектов. Поместив в дерево и настроив функциональные блоки, установите внутренние связи между ними. Для установления внешних связей лучше использовать отдельные переменные *Объекта* (используемого агрегата). Блоки помещаются в дерево следующим образом - найдите в палитре нужный функциональный блок, щелкните по нему левой клавишей мыши, переместите блок в дерево, щелкните левой клавишей еще раз в том месте, куда хотите вставить блок. Чтобы установить связь, нужно перетащить мышью, не отпуская левой клавиши, одну переменную к другой;
- На закладке *Опрос свойств объекта*, настройте параметры опроса агрегата;
- Создайте документы: мнемосхемы, тренды, и т.д. (закладки *Окна*, *Тренды* и *Другие*) - все, чем должен обладать агрегат. После этого можно еще составить расписание действий (закладка *Расписания*). При этом, если выполняется действие над документом, нужно выбрать в поле *Документ* из всплывающего списка один из уже созданных документов. Если же необходимо выполнить действие *Присвоить* над какой-либо переменной, то его надо перетащить мышью в поле *Документ* прямо из дерева.

Когда агрегат полностью готов к использованию, не забудьте поместить его в библиотеку.

Создание списка переменных

Функциональные блоки уже содержат предопределенный список входов и выходов. Для объектов создается необходимый набор входных (значения) и выходных (команды, события, расчеты) переменных и их групп. После создания переменных создаются их связи с источниками и приемниками данных. При создании переменных исходят из нескольких общих правил:

- *Разделение логических переменных от их физических источников*. Например, в мнемосхеме можно отобразить непосредственно значение OPC-переменной *Температура печи 1*. Однако, та же переменная может быть задействована не только в мнемосхеме, но и в других документах объекта *Печь 1*. Рациональнее создать в объекте переменную *Температура* и использовать ее во всех необходимых документах. Это облегчит тиражирование данной части проекта. Если нам нужна еще и *Печь 2*, то

после дублирования *Печи 1* останется только соединить переменную *Температура* с ОРС-переменной *Температура печи 2*. Кроме того, такое построение проекта гораздо оптимальнее с точки зрения быстродействия работы системы;

- **Правильный выбор уровня подчиненности переменных.** Пусть у нас имеется группа печей, каждая из которых представлена в дереве проекта объектом, имеющим мнемосхему. На мнемосхеме отображается значение суммарного значения расхода газа на все печи. Целесообразно переменную *Расход газа* включить в состав объекта *Группа печей*. Тогда при добавлении новой печи путем дублирования имеющейся не потребуется производить никаких действий для отображения значения этого параметра на мнемосхеме нового объекта;
- **Использование стандартных шкал.** Несмотря на то, что для переменной можно задать индивидуальную шкалу для контроля ее границ и отображения на трендах и в мнемосхемах, желательно все же при отсутствии необходимой шкалы в списке шкал создать новую шкалу, а не настраивать саму переменную. Это может пригодиться при появлении еще одного датчика с такой же шкалой, а также облегчит сопровождение уже созданной системы - проще все коррекции шкал производить в одном месте (такой подход особенно актуален в ситуации, когда создают систему одни люди, а сопровождают другие);
- **Правильный выбор имен переменных.** В ряде случаев при создании системы бывает необходимо для согласования с проектом автоматизации использовать кодовое обозначение сигнала в проекте. В то же время, для оператора удобнее работать с содержательным именем переменной. Рекомендуется в проекте MasterSCADA использовать то имя, которое планируется показывать оператору. Это позволит перетаскивать переменные из дерева проекта в документы без дополнительных действий по их переименованию. Для обеспечения согласованности с проектом автоматизации достаточно внести проектное обозначение в комментарий. При распечатке проекта MasterSCADA оба имени будут показаны рядом. В случае, если планируется использовать несколько переменных одного назначения, например, для отображения температуры в нескольких зонах печи, создайте одну с именем *Температура в зоне 1*, полностью настройте ее, а затем продублируйте, задав необходимое число копий. Новые переменные будут автоматически правильно пронумерованы, и не потребуют никакой настройки, кроме задания связи с источником данных;
- **Использование эффективных способов опроса.** Для редко изменяющихся переменных лучше задавать тип опроса *по изменению*. Опрос *по изменению* должен предусматривать *мертвую зону* для фильтрации случайных флуктуаций значения. Для переменных, используемых только в вызываемых документах, если они не предназначены для архивирования, лучше снять флажок *опрос при отсутствии потребителей*. Это сократит нагрузку на систему.

Настройка объектов

Для типовых элементов она не нужна, поскольку уже была сделана раньше. Но если в данной схеме есть какие-то особенности, типовой элемент можно подправить и сохранить в библиотеке с тем же или новым именем. Если же элемент не типовой, то надо описать все его свойства. Перемещаясь по закладкам свойств *Объекта*, задайте соответствующие настройки.

Параллельная разработка проекта

Проект может создаваться несколькими разработчиками на разных компьютерах *пообъектно*. То есть объекты создаются независимо (в рамках отдельных проектов), помещаются в библиотеки, библиотечные объекты копируются на один компьютер и из них составляется готовый проект. Обратите внимание, что пользовательские категории сообщений и интервалы значений являются настройками проекта, а не объектов. Если используется библиотечный объект, один из выходов которого имеет пользовательский интервал, а в проекте этот интервал не создан, то соответственно, он не будет установлен выводу. Аналогичная ситуация для тех сообщений, которые имеют категорию, отсутствующую в проекте. При копировании *Объектом* между проектами:

- **Диапазон изменения** (Интервал шкалы) переменных восстанавливается по имени Интервала;
- **Категории сообщений** для *Событий* и функциональных блоков, генерирующих сообщения, восстанавливаются по именам категорий;

- *Типы окон* объекта восстанавливаются по имени окна. При этом, если в новом проекте уже существует тип окна с тем же именем, то окну вставляемого объекта будут заданы стандартные размеры окон данного типа. Если же в проекте тип окна с таким именем отсутствует, то он будет создан.

Есть два пути решения этой проблемы: можно заранее договориться о том, какие интервалы и категории сообщений будут использоваться, и создать их в каждом проекте, в котором разрабатываются библиотечные объекты. Но надежнее поступить так: создать проект, в нем создать все необходимые интервалы и категории сообщений, затем скопировать этот проект на все компьютеры, на которых предполагается работать и использовать его как оболочку для создания объектов. Не забудьте, что, если на разных компьютерах создаются объекты, ссылающиеся друг на друга или на общие внешние объекты, при сборке проекта связи восстановятся, за исключением некорректных ссылок (на отсутствующие сигналы) и ссылок нескольких переменных-выходов на один и тот же *Вход ФБ* или *Значение*.

Тиражирование элементов проектов и использование библиотек

Для использования созданного элемента повторно возможно воспользоваться одним из трех возможных способов его тиражирования.

- **Копирование и вставка из буфера.** Скопируйте объект в буфер через контекстное меню, щелкните на том объекте, в который собираетесь вставить скопированный, выберите команду вставки из буфера;
- **Дублирование.** В случае, если планируется использовать несколько объектов или переменных одного назначения, например, для отображения температуры в нескольких зонах печи, создайте одну переменную с именем *Температура в зоне 1*, полностью настройте ее, а затем продублируйте, задав необходимое число копий. Новые переменные будут автоматически правильно пронумерованы, и не потребуют никакой настройки, кроме задания связи с источником данных;
- **Помещение в библиотеку и последующая вставка из нее.** Помещение производится через контекстное меню, вставка выполняется аналогично вставке функционального блока. Библиотечные объекты можно использовать при создании других проектов и при параллельной работе нескольких разработчиков с одним проектом. *Расчет* и *Событие* также можно помещать в библиотеку. При помещении в библиотеку они сохраняют все свои настройки и свойства.

Перенос библиотек на другие компьютеры. В папке для хранения библиотек и настроек, расположение которой определяется в диалоге настроек (его можно вызвать командой *Настройки* в меню *Сервис*), имеется вложенная папка *Library*, содержащая все библиотеки. Для переноса объекта (или библиотеки) на другой компьютер просто скопируйте его папку в папку *Library* другого компьютера и выполните команду *Обновить палитру*.

Восстановление связей при вставке объектов. При вставке объектов в проект они автоматически (если это не запрещено в настройках *Системы*) пытаются установить связи с внешними источниками и приемниками данных.

Особенности тиражирования элементов проектов при переносе из одного проекта в другой. При помещении в проект объектов из другого проекта существуют некоторые особенности, которые необходимо учитывать:

- Диапазон изменения (интервал шкалы) переменных восстанавливается по имени Интервала;
- Категории сообщений для событий и функциональных блоков, генерирующих сообщения, восстанавливаются по именам категорий. Если в новом проекте категории с аналогичными именами отсутствуют, то используется категория, заданная по умолчанию;
- Типы окон объекта восстанавливаются по имени окна. При этом, если в новом проекте уже существует тип окна с тем же именем, то окну вставляемого объекта будут заданы стандартные размеры окон данного типа;
- Привязка объектов к компьютерам в дереве системы не восстанавливается;
- Дополнительные настройки журналов объектов, заданные при работе с ними в режиме исполнения в другом проекте, при переносе объекта в новый проект сбрасываются;

- При вставке компьютера из другого проекта он теряет все настройки, унаследованные им от системы в первоначальном проекте, и приобретает все настройки, заданные для системы в новом проекте. Вставленные в компьютер OPC-серверы сохраняются.

Объектная идеология

Инструментальная среда MasterSCADA предлагает объектный подход к разработке проекта. Объект в MasterSCADA - это основная единица разрабатываемой системы, соответствующая реальному технологическому объекту (цеху, участку, аппарату, насосу, задвижке, датчику и т.п.), управляемому разрабатываемой с помощью MasterSCADA системой. С другой стороны, это и традиционный с точки зрения программирования объект, обладающий стандартными для программных объектов качествами. Объект имеет набор свойств и документов, которые жестко связаны с ним.

Свойства объекта – это, например, период опроса и способ обработки сигналов от его датчиков. Документы объекта – его изображение, описание, чертеж, перечень сообщений и т.п. В MasterSCADA нет просто тренда, рапорта или мнемосхемы: каждый документ в разрабатываемой системе всегда относится к какому-либо объекту, являясь его свойством. По умолчанию все настройки наследуются от родительского объекта.

Каждый объект имеет множество настроек. Такое обилие могло бы потребовать от разработчика системы выполнения огромного количества действий. Но так как для разных объектов их список в основном одинаков, то все настройки можно сделать только один раз, все подчиненные объекты автоматически воспримут настройки родительского элемента, то есть унаследуют их. Исключение будут составлять только те настройки и только у тех элементов, которые разработчик изменил сам. Схема наследования свойств представлена на следующем рисунке.



Допускается многократное использование одного и того же объекта со всеми созданными для него документами, в том числе при разработке различных систем. При копировании объекта или сохранении его в библиотеке все его настройки и документы, настройки документов и внутренние связи будут сохранены. Внешние связи с источниками данных будут восстановлены при наличии источников с такими именами, внешние связи с приемниками данных будут восстановлены, если эти приемники данных свободны, остальные будут показаны в общем списке. Благодаря этому управление и контроль типовым технологическим объектом (насосом, задвижкой, реактором, фильтром и т.п.) создаются один раз для всех проектов. Это позволяет создавать объекты для одной системы параллельно независимыми разработчиками.

Создание связей объекта с OPC переменными

Связывать переменные Объекта с OPC переменными можно двумя способами:

- Если в *Объекте* уже есть переменная, которую вы хотите связать с OPC переменной, то перетащите мышью одну к другой (также, как делается связь между двумя переменными объекта);
- Если необходимо создать сразу несколько связей, то не обязательно предварительно вставлять переменную объекта. Возможно выделить несколько переменных в дереве *Системы* и перетащить их в дерево объектов. При этом в *Объекте* автоматически будут созданы переменные с названиями, максимально соответствующими названиям OPC переменных (без конфликтов имен), у которых уже будет настроена связь.

Устранение неполадок:

- **OPC сервер не виден из клиента.** Необходимо проверить, что *opc enum* правильно настроен и работает в данный момент. Если проблему не удалось устранить, то необходимо повторить установку *opc enum*, используя пакет *OPC Foundation - OPC Core Components 2.00 Redistributable*;
- **OPC сервер виден, но невозможно вставить его переменные.** Проверить загружаемую по умолчанию конфигурацию OPC сервера. Необходимо её настроить с помощью средств OPC сервера. Проверить, что данный сервер зарегистрирован в операционной системе и его *dcom* компонент настроен в соответствии с инструкцией. Для OPC сервера ИнСАТ необходимо запустить утилиты *unregister*, а затем *register*. Это приведёт к повторной регистрации OPC сервера;
- **OPC сервер виден, но не запускается.** Если клиент выдаёт сообщение: *Не возможно запустить OPC сервер на удалённой машине*, то необходимо повторить регистрацию OPC сервера и регистрацию этой библиотеки *opcproxy.dll*, которая расположена в папке *%system32%* операционной системы *Windows*;
- **OPC сервер виден и переменные вставлены, но в режиме работа значения переменных отсутствуют.** Проверить, что в настройках *dcom* типа *подключения*, установлено *нет*. Проверить связь с прибором, возможна проблема с драйвером устройства. Смотрим отладочную информацию по OPC серверу (есть или нет ответа от прибора).

Определение периода опроса

Страница свойств *Опрос сервера OPC*:

The screenshot shows the 'OPC Server Query' properties dialog box with the 'Query' tab selected. The dialog is divided into several sections:

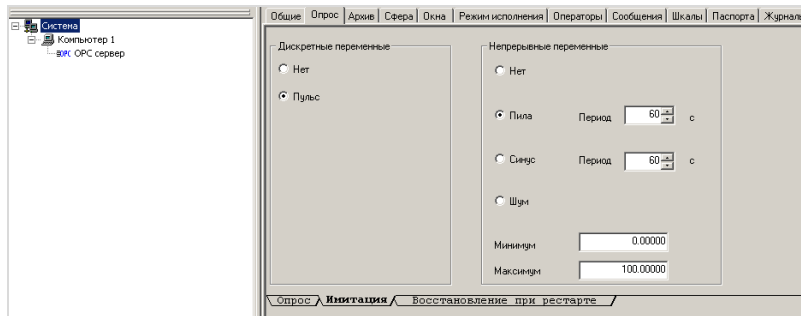
- Периодический опрос изменений (Periodic query of changes):**
 - Периодический опрос изменений
 - Период: 1000 мс
 - 'Мертвая зона': 0.00000 %
 - Старт по условию: [text box]
- Полный опрос (Full query):**
 - Период: 120000 мс
 - Перезапуск OPC сервера при отсутствии ответа в течении: 3 периодов
 - Опрос по условию: [text box]
 - Использовать синхронный опрос
- Передача данных OPC серверу (Data transfer to OPC server):**
 - Периодическая: Период записи данных: 1000
 - По изменению входов: 'Мертвая зона': 0.00000 %
 - 0.00000
 - Возврат к умолчанию
 - Запрет использования некачественных значений
 - Устанавливать локальную метку времени
- Переподключаться по условию: [text box]

MasterSCADA принимает значения переменных из сервера OPC при их изменении:

- MasterSCADA получает значения из сервера OPC не чаще, чем величина периода (задается в группе *Периодический опрос изменений*);

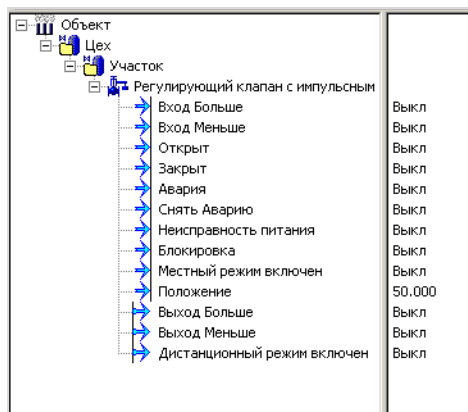
- Сигнал принимается, если он изменился больше, чем на величину мертвой зоны (задается в группе *Передача данных OPC серверу*);
- Период (задается в группе *Полный опрос*), с которым MasterSCADA принудительно запрашивает значения, даже если они не изменились.

Использование имитации:

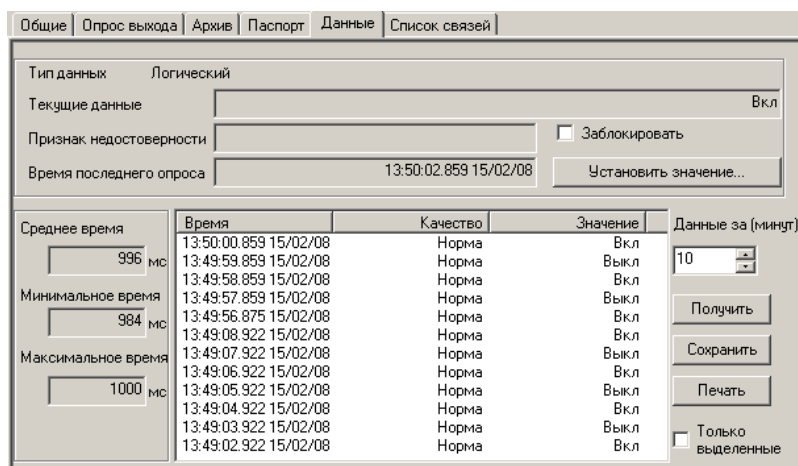


В режиме имитации на Входы без связей подаются имитационные сигналы. Здесь есть возможность настроить законы, по которым эти сигналы будут изменяться. Обратите внимание, что дискретные команды будут сменять друг друга не с заданным периодом, а с периодом опроса.

Значения Входов и Выходов видны прямо в дереве объектов. Можно увидеть значение любой переменной без каких-либо специальных действий как показано на рисунке ниже.



Чтобы управлять значением, следует выделить *Вход* или *Выход*, открыть закладку *Данные*.



- В поле с данными можно просмотреть архив переменной с метками времени;
- Если необходимо, чтобы значение не изменялось, то следует поставить флаг в поле *Заблокировать*;
- В поле *Текущие данные* отображаются текущее значение;
- Чтобы установить свое значение, следует нажать на кнопку *Установить значение*.

Чтобы выяснить оптимальный период опроса, необходимо:

- Для OPC сервера на закладке *Опрос* выставить период меньше ожидаемого;

Общие | Опрос | Диагностика

Периодический опрос изменений

Период мс

"Мертвая зона" %

Старт по условию

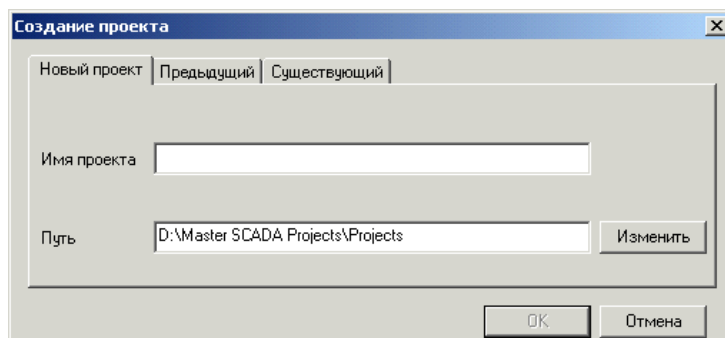
- В режиме исполнения для переменной открыть закладку *Данные* и проследить среднее время. Это время и нужно будет выставить в качестве периода опроса.

Время последнего опроса

	Время	Качество
Среднее время <input type="text" value="998"/> мс	14:07:09.234 15/02/08	Норма
	14:07:08.234 15/02/08	Норма
	14:07:07.234 15/02/08	Норма
Минимальное время <input type="text" value="984"/> мс	14:07:06.234 15/02/08	Норма
	14:07:05.234 15/02/08	Норма
	14:07:04.250 15/02/08	Норма
Максимальное время <input type="text" value="1000"/> мс	14:01:30.000 15/02/08	Норма
	14:01:29.000 15/02/08	Норма
	14:01:28.000 15/02/08	Норма
	14:01:27.000 15/02/08	Норма
	14:01:26.000 15/02/08	Норма
	14:01:25.000 15/02/08	Норма

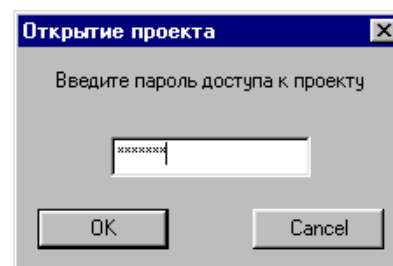
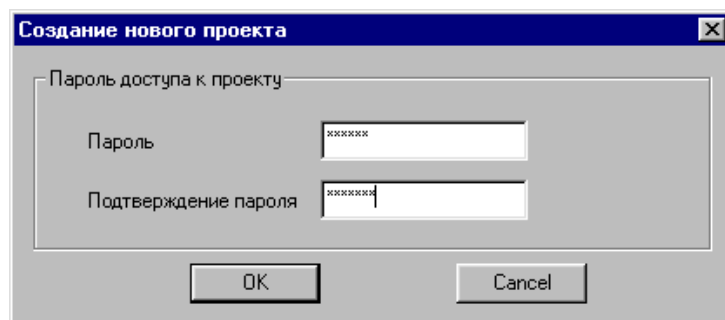
Работа с проектом

При запуске программы или нажатии кнопки **Создать** на экран выводится окно



В окне можно сделать одно из следующих действий:

- **Создать новый проект**
Введите имя нового проекта в поле **Имя проекта** и нажмите кнопку **ОК**. Если необходимо создать проект в папке, отличной от папки для проектов по умолчанию (путь к ней указан ниже), нажмите **Изменить** или введите путь вручную. При создании нового проекта предлагается ввести пароль, который впоследствии необходимо будет вводить для открытия.



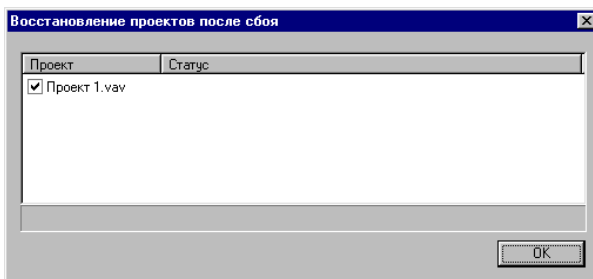
- **Открыть один из последних проектов**, который редактировался на этом компьютере. Для этого нужно просто перейти на закладку **Предыдущий** и выбрать проект
- **Открыть любой существующий проект**
Для этого надо перейти на закладку **Существующий**. В списке выбрать любой проект, находящийся в папке по умолчанию (список отображает 50 проектов, если проект отсутствует в списке, но имеется на компьютере нажмите **Обзор**). Если проект был перемещён в другую папку, нажмите кнопку **Обзор** и найдите файл проекта самостоятельно. Открыть один из последних проектов или любой существующий проект можно с помощью кнопки **Открыть**. В зависимости от выбора команды **Открыть** или **Создать** диалог открывается с позиционированием на нужной закладке.
- **Удалить проект**
Для удаления проекта выберете проект из закладки **Существующий** и нажмите кнопку **Удалить**.

Экспорт проектов осуществляется командой **Экспортировать** из меню **Проект**. Эта команда создает копию проекта. С помощью нее удобно переносить проект между компьютерами. В раскрывающемся диалоговом окне нужно выбрать папку назначения для экспортируемого проекта (это может быть как локальная папка, так и расположенная на другом компьютере). При экспорте проектов копируются все необходимые проектные файлы, в том числе ресурсы мнемосхем, что гарантирует полную переносимость проекта. Файлы архивов объектов при экспорте проекта не копируются.

Импорт проекта сохраняет под именем текущего проекта тот проект, который выбран в поле **Папка с импортируемым проектом**. Если импортируемый проект был изменен в режиме исполнения, то для сохранения этих настроек необходимо установить соответствующий флаг.



Если при работе с проектами программа была некорректно завершена (например, из-за сбоя питания компьютера), при следующем запуске будет предложено восстановить последнее зафиксированное состояние проекта(ов).



Для отказа от восстановления нужно снять переключатель слева от имени проекта или просто закрыть это окно. Тогда проект будет находиться в том виде, в котором его последний раз сохраняли. Состояние проекта запоминается после каждых семи действий, а также через определенные промежутки времени (возникает надпись **Сохранение Undo-стека** в строке состояния).

Элементы дерева системы

В **дереве Системы** настраиваются элементы проекта, которые отвечают за распределение выполняемой задачи между имеющимися компонентами. В программе реализована возможность производить распределенное управление технологическим процессом, что предусматривает использование нескольких компьютеров для вычислений (пересчета функциональных блоков) и обмена данными с аппаратными средствами (с помощью OPC серверов). Кроме того, в MasterSCADA реализована возможность взаимодействия с внешними базами данных, посредством так называемых БД-коннекторов. Размещение в проекте компьютеров, подключение БД-коннекторов и OPC серверов - этапы создания системы управления, выполняемые в дереве Системы.

Компьютеры

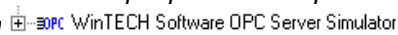
Проект создается на одном **Компьютере**, но в режиме исполнения задачи могут быть распределены, и выполнение части задач может происходить на разных физических компьютерах. Для этого необходимо поместить **Компьютер** в **дереве Системы**, при этом можно задать ему уникальное имя. Совершенно не нужно задавать **Компьютеру** сетевое имя - лучше выбрать имя, соответствующее его роли в структуре АСУТП. При старте режима исполнения появится окно, в котором выбирается **Компьютер**, который ассоциируется с данной машиной по заданному имени. Имена всех Компьютеров автоматически помещаются в раскрывающийся список на закладке **Общие** у свойств каждого объекта, где можно выбрать тот **Компьютер**, который должен заниматься управлением объектом.

При распределении **Компьютеров** для объектов необходимо учитывать, что обмен данными между объектами на одном компьютере осуществляется быстрее, чем между объектами на разных **Компьютерах**, поэтому при назначении компьютера для того или иного объекта в первую очередь следует обратить внимание на его связи с другими объектами и OPC серверами.

OPC серверы и OPC переменные

OPC сервер - программа, которую MasterSCADA использует в качестве *посредника* для доступа к данным, поступающим от аппаратных средств. Это можно представить следующим образом:



- ❖ **Важно!** Есть различие между понятиями OPC сервер контроллера (программа) и OPC сервер - элемент дерева системы. Первое - это самостоятельная программа, смысл которой описан выше. Второе - объект, предназначенный для удобства использования этой программы в дереве Системы, где он выглядит примерно таким образом 

Для обоих понятий используются одинаковые термины, о какой конкретно сущности идет речь всегда будет ясно из контекста. Для того, чтобы использовать OPC сервер контроллера необходимо реализовать следующие действия:

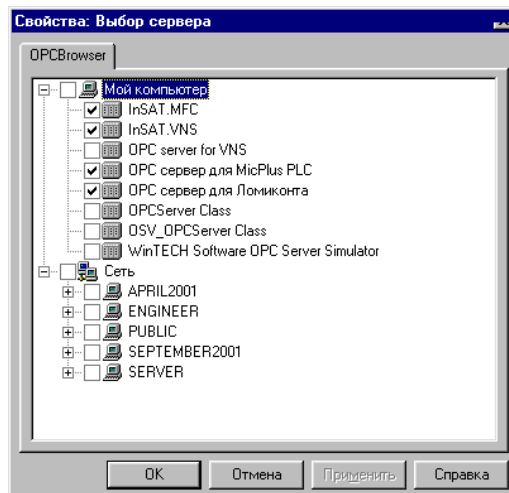
- Выполнить команду **поиск OPC серверов** (из **контекстное меню Компьютера**, или меню **Добавить**), если она не была выполнена раньше;
- **Вставить** необходимые OPC серверы (функциональные блоки) в **Компьютер**. Это можно сделать одним из трех способов: через контекстное меню Компьютера; из списка OPC серверы в меню **Добавить**; из библиотеки OPC серверы в палитре элементов;
- Вставить OPC переменные.

OPC HDA сервер - программа, которую MasterSCADA использует в качестве *посредника* для доступа к архивным данным, которые предоставляет контроллер или счетчик коммерческого учета. Контроллер (счетчик) накапливает эти данные путем опроса аппаратных средств.

Все полученные данные от OPC HDA сервера MasterSCADA сохраняет в своем архиве. Дальнейшая работа с этими данными осуществляется при помощи стандартных функций MasterSCADA. Возможно просматривать эти данные при помощи трендов. При переходе в режим исполнения MasterSCADA получает все накопленные данные с момента выхода из режима исполнения, а также продолжает получать новые данные по мере их поступления.

Диалоговое меню поиска OPC DA или HDA серверов вызывается по команде главного меню **Сервис - Органайзер - Поиск OPC Серверов** или через контекстное меню **Компьютера**.




В результате на экране появится органайзер приблизительно такого вида:



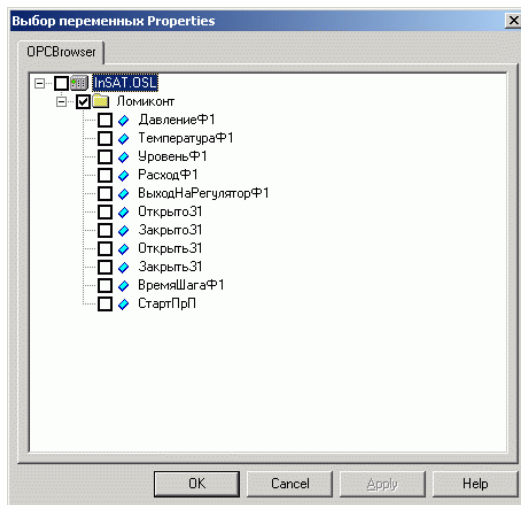
В окне отображено дерево OPC серверов, обнаруженных на данном компьютере и в ближайшем сетевом окружении. Отметьте флажками OPC серверы, которые собираетесь использовать в проекте и впоследствии они будут отображены в **контекстном меню Ком-**

пьютера, раскрывающейся команде **OPC серверы** в меню **Добавить**, а также в **категории OPC серверы** в палитре элементов.

Доступ к данным OPC серверов в MasterSCADA осуществляется через OPC переменные. Существует три основных вида OPC переменных:

- Для чтения (отображается в дереве значком выхода );
- Для записи (отображается в дереве значком входа );
- Для чтения и записи (отображается в дереве значком .

Чтобы поместить в **Проект** OPC переменные щелкните правой кнопкой мыши на OPC сервере, в контекстном меню выберите пункт **Вставить** и нажмите на **OPC переменные**, предполагается, что они в OPC сервере уже настроены (способ настройки переменных зависит от конкретного OPC сервера контроллера и описан в его собственной документации). В результате этого действия появится диалог **Выбор переменных** содержащий дерево, в котором отображены OPC переменные:



Здесь следует отметить флажком те переменные или целые группы, которые предполагается использовать в проекте и нажать **OK**. После этого в **дереве Системы** будут сразу отображены выбранные группы и переменные, с которыми можно совершать те же действия, что и с входами и выходами функциональных блоков, то есть перетаскивать мышью для установления связей, изменять параметры страницы свойств и т.д. Если предполагается использовать в **Проекте** все **Переменные** OPC сервера, выберите в контекстном меню OPC сервера пункт **Вставить все** и в дереве отобразятся сразу все созданные группы и переменные. В некоторых OPC серверах имена групп переменных совпадают с переменными находящимися на том же уровне. В этом случае предусмотрено следующее:

- При вставке переменной, имя которой совпадает со вставленной ранее группой выдается сообщение в протоколе работы и переменная не вставляется;
- При вставке группы, имя которой совпадает со вставленной ранее переменной выдается сообщение в протоколе работы и группа не вставляется;
- При вставке переменной и группы с одинаковыми именами вставляется только переменная и выдается сообщение о невозможности вставить группу.

В MasterSCADA реализована поддержка **OPC переменных**, являющихся массивами. При вставке такой переменной создается группа (по имени переменной), в которую добавляются OPC переменные, соответствующие элементам массива. Количество элементов в массиве устанавливается на странице свойств группы OPC переменных **Общие**. Имена переменных в группе соответствуют индексу элемента в массиве, т.е., в режиме исполнения на элемент группы с именем **1** поступает значение из массива по индексу **1**.

- ❖ **Важно!** OPC переменные можно не только связывать с уже существующими переменными дерева объектов. Несколько выделенных OPC переменных (используйте групповое выделение) можно таскать прямо в Объект. При этом в Объекте автоматически создаются Входы-Выходы, уже связанные с этими OPC переменными.

БД-коннекторы и блоки процедур

Кроме взаимодействия с OPC-серверами, в MasterSCADA реализована возможность обмена данными с *верхним уровнем* - базами данных. На сегодняшний день поддерживаются следующие серверы баз данных: MS SQL Server, Oracle, Sybase, Interbase, а также MySQL.

MasterSCADA поддерживает несколько механизмов использования баз данных. Для получения и передачи данных можно использовать собственные хранимые процедуры, созданные в самой БД. В этом случае в дереве системы организация взаимодействия между MasterSCADA и базой данных осуществляется с помощью БД-коннекторов, блоков процедур и блоков функций.

БД-коннекторы - отвечают за подключение к базе данных. БД-коннекторы в своем составе могут содержать Блоки процедур и Блоки функций, представляющие хранимые процедуры БД.

Блоки процедур и **Блоки функций** - позволяют выполнять хранимую процедуру или функцию базы данных, доступную через данный **БД-коннектор**. Блок процедуры представлен в дереве системы как функциональный блок, имеющий набор групп входов и выходов, соответствующих его настройкам. Блок процедур может иметь следующие переменные в дереве:

- Группа **Входные параметры**. Содержит входы, соответствующие входным и входным/выходным параметрам блока процедуры или функции;
- Группа **Выходные параметры**. Содержит выходы, соответствующие выходным и входным/выходным параметрам Блока процедуры или функции;
- **Выход Результат**. Соответствует результату выполнения хранимой процедуры. В MSSQL результат могут возвращать все процедуры. В Oracle результат могут возвращать только функции. Наличие данного выхода зависит от настроек Блока процедуры;
- Группа выходов **Выборка**. Соответствует первой записи resultset'a, возвращаемого хранимой процедурой.

Контроллеры и модули ввода/вывода

Под **Контроллером** подразумевается вычислительное устройство, входящее в состав АСУТП и выполняющее одну из следующих функций:

- Сбор информации о технологическом процессе;
- Расчет косвенных параметров, управляющих воздействий, интегралов и др.;
- Выдача команд и управляющих воздействий на исполнительные механизмы;
- Обмен информацией с другими вычислительными устройствами (контроллерами или компьютерами).

При всем многообразии имеющихся на рынке контроллеров (по внешнему виду, степени устойчивости к неблагоприятным воздействиям и др.) все контроллеры имеют в своем составе **Модули ввода/вывода** и **Коммуникационные порты**. Помимо этих обязательных устройств контроллеры могут быть оснащены дисками или другими средствами хранения информации, индикаторами, клавиатурами, мониторами и другими устройствами, облегчающими процесс разработки программ, настройки и сопровождения Контроллеров или обеспечивающих выполнение специфических функций АСУТП.

Модули ввода/вывода служат посредниками между технологическим оборудованием и вычислительными устройствами.

Модули ввода трансформируют электрические сигналы, поступающие от датчиков, в цифровую форму, доступную для обработки в вычислительных устройствах. По типу поступающей информации модули ввода бывают аналоговыми (преобразуют сигналы от датчиков температуры, давления, расхода и других физических величин) и дискретными (преобразуют сигналы от датчиков состояния, например: насос включен или выключен, дверь открыта или закрыта и т.п.). Это наиболее распространенные модули ввода. Существуют и другие. Так счетчики импульсов или частоты по дискретному сигналу вычисляют его аналоговую характеристику.

Модули вывода преобразуют команды и управляющие воздействия, рассчитанные контроллерами, в электрические сигналы для исполнительных механизмов. Они тоже бывают аналоговыми или дискретными.

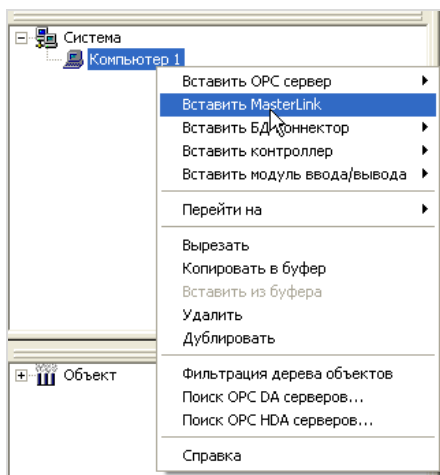
Цифровая форма сигнала представляет собой двоичный код определенного диапазона. Для удобства работы с сигналами модулей в MasterSCADA реализовано масштабирование. При масштабировании диапазону кода ставится в соответствие диапазон изменения физической величины. В результате чего пользователь работает не со значениями переменных в коде преобразователя, а со значениями, соответствующими реальным физическим величинам. Диапазоны изменения величин задаются на закладке свойств системы **Шкалы**, затем выбираются на закладке Общие для каждой переменной модуля.

MasterLink

MasterLink – это шлюз для опроса устройств, подключенных к компьютеру. Используются только последовательные порты. Данный модуль можно использовать как альтернативу OPC серверам.

❖ **Важно!** *MasterLink является упрощенной версией Windows-контроллера.*

Модуль MasterLink добавляется через контекстное меню компьютера.



Настройки MasterLink:

Горячий рестарт	
Период сохранения данных (с)	1
Время актуальности данных (с), 0 - всегда	0
Разное	
Тип	MasterLink
Количество последовательных портов	2
Управление выполнением	
Отображение отладочных сообщений	В окне
Чтение	
Период опроса	1000

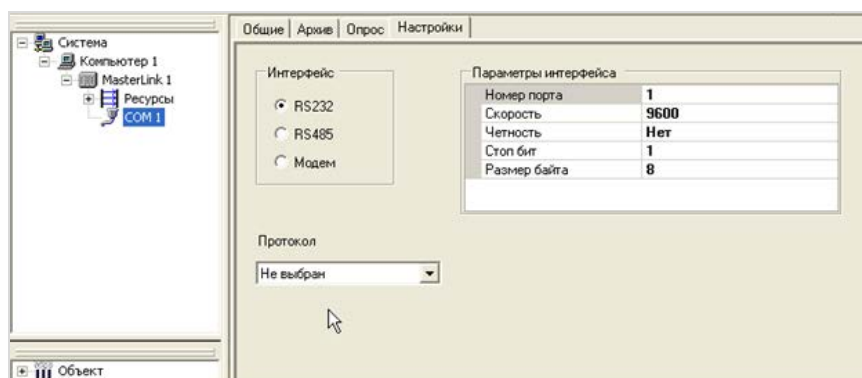
При помощи настройки **Количество последовательных портов** задаётся количество задействованных COM портов в системе. Настройка Отображения отладочных сообщений позволяет задать тип вывода сообщения:

- **В окне** – создается отдельное окно с отображением сообщений;
- **Запись в файл** – сообщения записываются в отдельный файл;
- **Не отображать** – отладочные сообщения не отображаются и не записываются.

Период опроса – это период получения данных компьютером из модуля MasterLink. В MasterLink включено архивирование в контроллере, поэтому если COM порт опрашивается с периодом в 100 мс, а MasterLink с периодом в 1с, то при включенном архивировании потери данных не будет.

Настройки обмена последовательного порта

- ❖ **Важно!** *Описание настройки протоколов для шлюза MasterLink, подходит и для контроллеров под управлением MasterPLC.*



Настройки группы Параметры интерфейса:

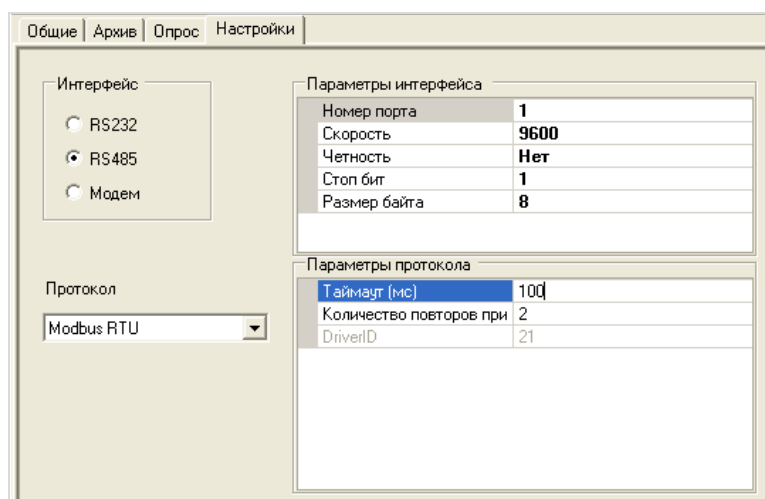
- **Номер порта** – задание номера COM порта, к которому подключено опрашиваемое устройство;
- **Скорость** – скорость обмена по интерфейсу;
- **Четность** – наличие/отсутствие контроля четности (Да – Нет);
- **Стоп бит** – количество используемых стоповых бит (значения 1, 1.5, 2);
- **Размер байта** – количество бит в посылке (значения 7 или 8).

Интерфейсы передачи данных в данном окне свойств бывают трёх типов:

- **RS-232.** Интерфейс для подключения одного прибора. Поддерживает связь с верхним уровнем, драйвер прозрачного канала. Возможно подключение устройств Danfoss ECL, Меркурий 230, Логика (961, 762, 942), Пульсар, ПЧ Mitsubishi;
- **RS-485.** Интерфейс для подключения нескольких приборов. Поддерживает связь с верхним уровнем, драйвер прозрачного канала, ФБ-протокол Smartlink. Возможно подключение аналогичных RS-232 типов устройств, помимо этого поддержан ADAM-совместимый протокол и протокол Modbus RTU;
- **Модем.** Интерфейс для подключения GSM-модема. Поддержана только связь с верхним уровнем.

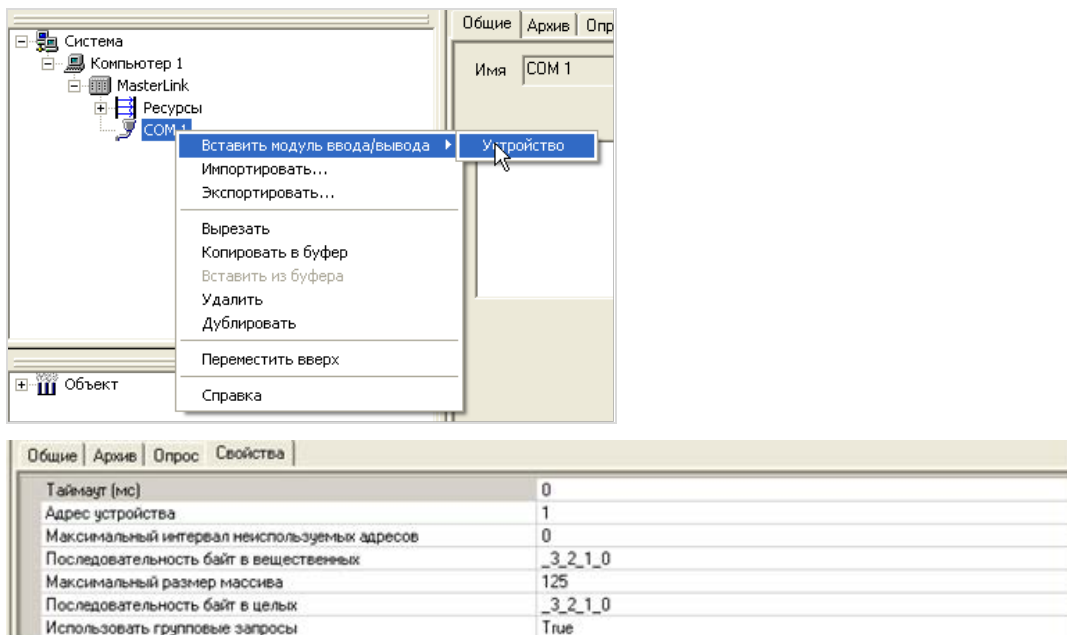
В группе **Протокол** COM порту присваивается протокол, по которому будет вестись обмен. После выбора протокола появляется дополнительное окно **Параметры протокола**.

Протокол **Связь с верхним уровнем** предназначен для реализации обмена между верхним уровнем (MasterSCADA или MasterPLC-OPC Server) и контроллером под управлением MasterPLC. Протокол **Modbus** – стандартный открытый протокол для организации связи электронного оборудования, один из самых распространенных в промышленности.



В **Параметрах протокола** может быть настроен **Таймаут** – в течение этого времени MasterLink ожидает ответ от устройства. В случае если от устройства не пришел ответ, или ответ был забракован, предпринимается попытка повторного запроса. **Количество повторов при неудачном запросе** – если через заданное количество повторов от устройства не получен корректный ответ, то у устройства устанавливается флаг **Отказ**.

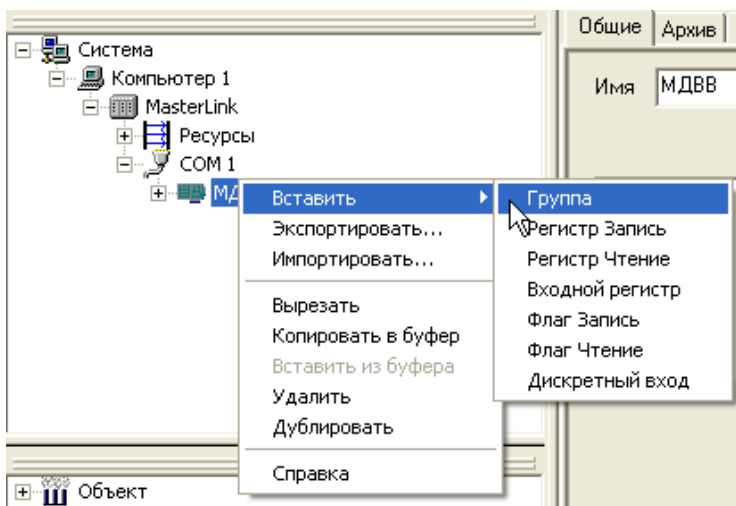
Добавление устройства в порт осуществляется через контекстное меню нужного порта. В случае Modbus RTU, необходимо добавить устройство и добавить в него нужные регистры чтения/записи.



Настройка параметров устройства происходит на закладке Свойства, как показано на картинке выше:

- **Таймаут** - в течение этого времени MasterLink ожидает ответ от устройства. По истечению этого времени предпринимается попытка повторного запроса. Если установлен 0, то время таймаута наследуется от настроек COM порта;
- **Адрес устройства** – адрес устройства в сети Modbus. Задается в десятичном формате;
- **Максимальный интервал неиспользуемых адресов** – если адреса опрашиваемых регистров идут с перерывами, позволяет оптимизировать запросы (посылается один групповой запрос, если перерывы в адресах не более этого значения);
- **Последовательность байт в вещественных (целых)** – при посылке по Modbus данных в 4 байта возможны различные варианты кодирования (*младшим вперед*, *старшим вперед* и т.д.). При помощи данного параметра можно подстроиться под конкретный прибор;
- **Максимальный размер массива** – максимальное количество запрашиваемых регистров при групповом опросе;
- **Использовать групповые запросы** – разрешает или запрещает использование одного запроса для опроса нескольких идущих подряд регистров.

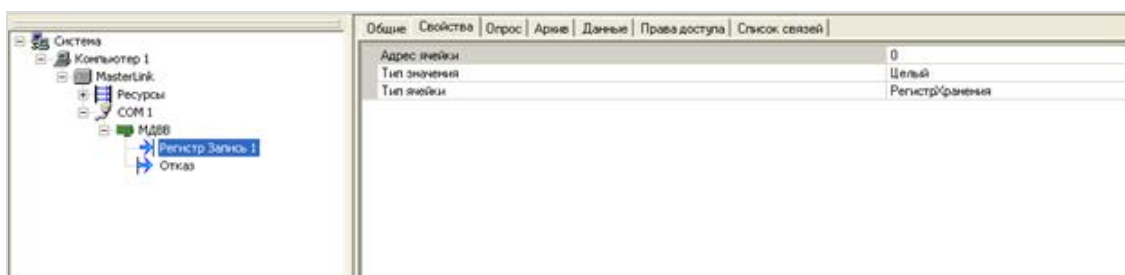
Добавление **Регистров** осуществляется через контекстное меню устройства.



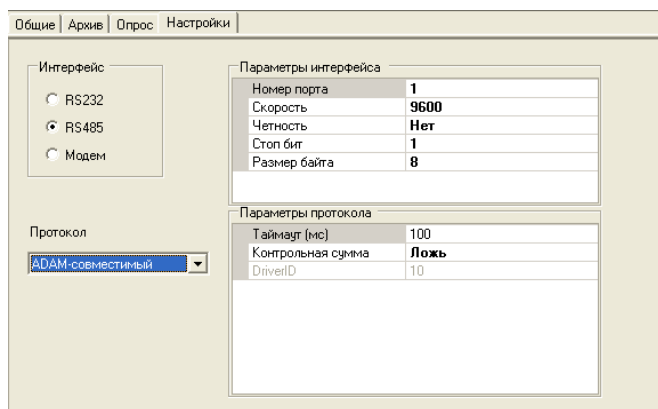
Описание добавляемых элементов в таблице ниже.

Обозначение в MasterSCADA	Функция Modbus	Описание
Регистр запись	16 (0x10) Preset Multiple Registers	Запись значений в несколько регистров хранения
Регистр чтение	3 (0x03) Read Holding Registers	Чтение значений из нескольких регистров хранения
Входной регистр	4 (0x04) Read Input Registers	Чтение значений из нескольких регистров ввода
Флаг запись	5 (0x05) Force Single Coil	Запись значения одного флага
Флаг чтение	1 (0x01) Read Coil Status	Чтение значений из нескольких регистров флагов
Дискретный вход	2 (0x02) Read Discrete Inputs	Чтение значений из нескольких дискретных входов

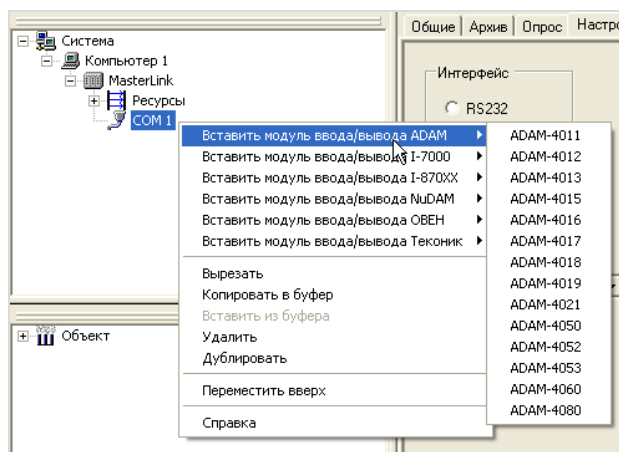
Вставляемые элементы имеют настройки, которые задаются на закладке Свойства. Адрес ячейки – это адрес регистра в приборе, который задаётся в десятичном формате.



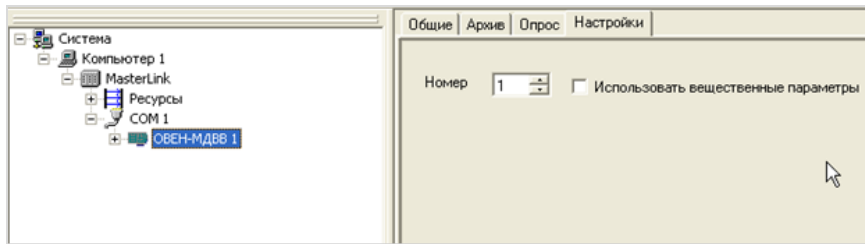
Для настройки протокола открытого протокола обмена по сети RS-485 – **DCON**, который был разработан компанией Advantec и применяется для обмена данными с модулями ввода/вывода ADAM, ICP DAS и другими (Овен, Текон, Контравт), необходимо задать следующие настройки.



Если в течение заданного времени в параметре **Таймаут** от устройства не приходит ответ, то формируется сигнал ошибки – **Вкл** на выходе **Отказ** и всплывающее соответствующее информационное сообщение. Для добавления устройства работающего по протоколу DCON в COM порт необходимо открыть соответствующее контекстное меню, как на картинке ниже, и выбрать необходимое из предоставленного списка.

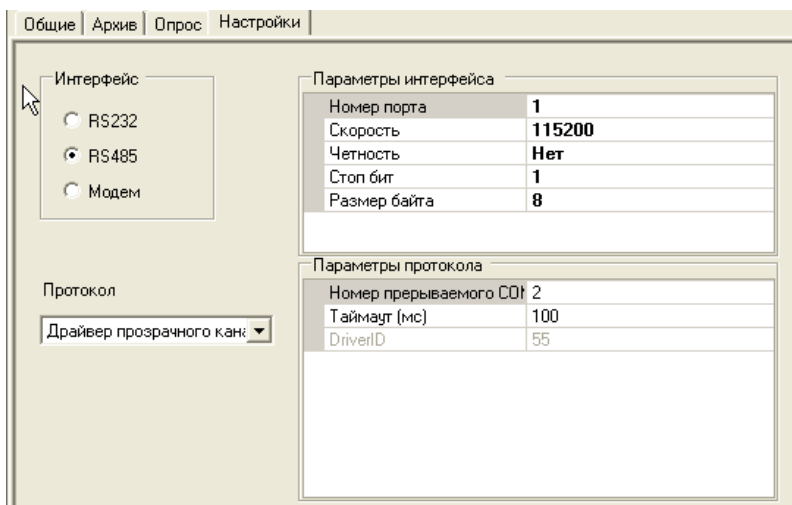


В добавленном устройстве, на закладке **Настройки** задается номер, который имеет прибор на шине.



Таким образом, для настройки устройств с протоколом **DCON** необходимо установить проверку контрольной суммы, добавить нужный модуль в COM порт и задать его адрес.

Драйвер прозрачного канала позволяет получать доступ к прибору из внешних устройств по интерфейсу RS-232/485.

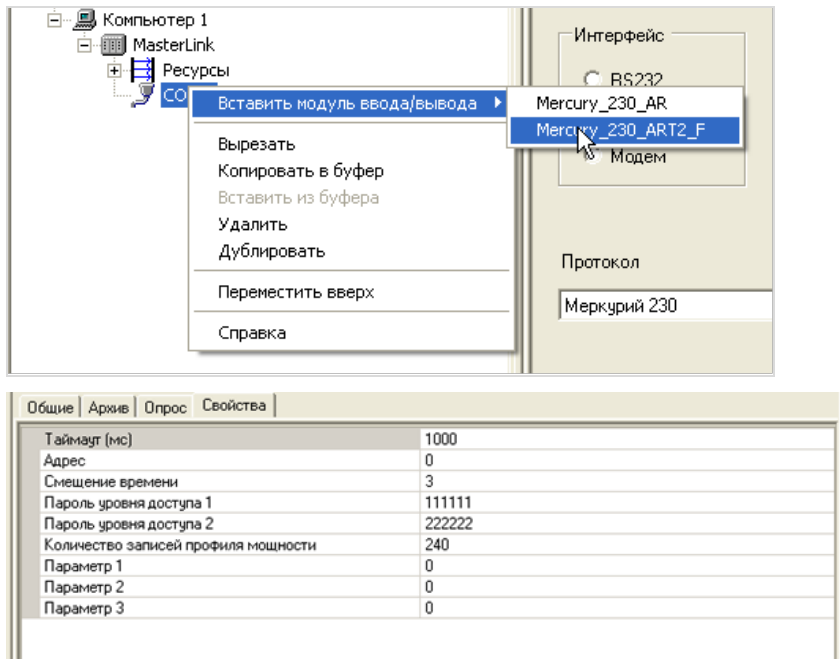


Протокол имеет следующий принцип работы - когда на COM-порт поступают данные, то происходит прерывание передачи данных порта указанного в настройке **Номер прерываемого СОМ порта**, и в этот порт пересылаются полученные данные.

- **Пример:** К контроллеру подключено устройство, работающее по протоколу Modbus RTU, но с ним периодически необходимо работать из другого устройства (например, компьютера). Устройство с Modbus RTU подключается к порту COM1 контроллера, настройка протокола – Modbus RTU. Компьютер подключается к устройству COM2 контроллера, настройка протокола – драйвер прозрачного порта, Номер прерываемого СОМ порта - 1. Когда на COM2 поступят данные, обмен COM1 - устройство будет прерван, и устройству будут посланы данные, которые поступили на COM2. Далее драйвер ожидает ответа в течение времени равного параметру Таймаут, направляет полученный от устройства ответ в порт, на который назначен драйвером прозрачного порта (COM2), и освобождает прерванный порт (COM1).

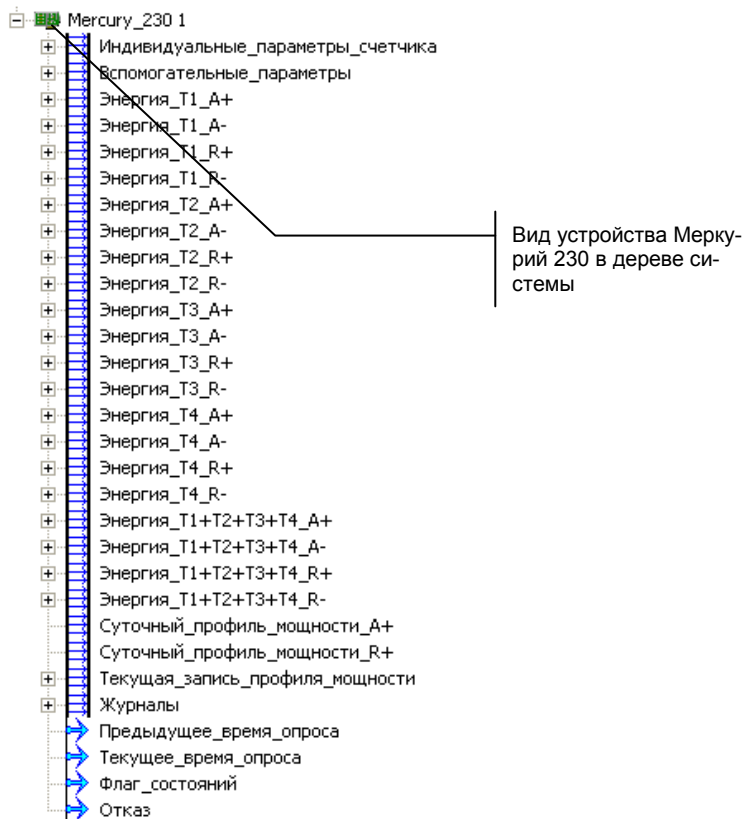
Протокол ФБ используется при работе через **Smartlink**. Подробное описание работы данного функционального блока описано в последующих главах.

Протокол **Меркурий 230** предназначен для получения данных от счетчиков электроэнергии Mercury_230_AR и Mercury_230ART2_F. Счетчики предназначены для учета активной и реактивной электрической энергии и мощности, в трехфазных 3-х и 4-х проводных сетях переменного тока частотой 50 Гц через измерительные трансформаторы тока или непосредственно, с возможностью передачи измерений и накопленной информации об энергопотреблении по цифровым интерфейсным каналам. Меркурий 230 AR – однонаправленный однотарифный счетчик, без поддержки архива профиля мощности и событий. Меркурий 230 ART2_F – двунаправленный четырехтарифный счетчик, с поддержкой архива профиля мощности и журналом событий. Счетчики добавляется через контекстное меню COM порта.



Свойства прибора (для прибора Меркурий 230 ART2_F):

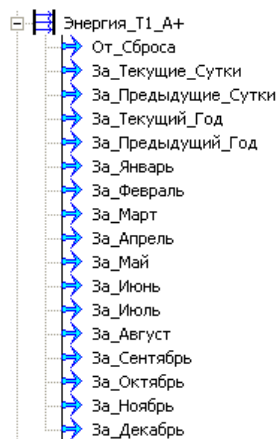
- **Таймаут** – Работает аналогично таймауту прокола DCON;
- **Смещение времени** – задание смещение времени в часах (задание часового пояса);
- **Пароль уровня доступа 1** – определяет разрешение на исполнение счётчиком команды считывания энергетических и вспомогательных параметров;
- **Пароль уровня доступа 2** – определяет разрешение на исполнение счётчиком команды по смене тарифов и программирования параметров счётчика на уровне энергосбыта;
- **Количество записей профиля мощности** – при старте контроллера (или MasterLink) происходит считывание заданного количества записей из профиля мощности;
- Параметр 1, Параметр 2, Параметр 3 – зарезервировано.



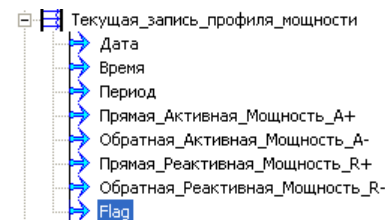
В группе **Индивидуальные параметры счетчика** содержатся параметры счетчика: Сетевой_адрес, Серийный_номер, Дата_выпуска, Местоположение. Группа **Вспомогательные параметры** - содержит действующие параметры сети: Напряжение, ток, частота сети, активная, реактивная, полная мощность на каждой фазе, угол между фазами, коэффициенты мощности. В группах **Энергия** содержится накопленное значение энергии.

- **Пример расшифровки значений:** группа *Энергия_T1_A+*, где: *T1* – номер тарифа, *A (P)* – активная или реактивная энергия, + (-) – прямое или обратное направление энергии (в случае использования двунаправленного счетчика).

Группы **Энергия_T1+T2+T3+T4_A+/- или R+/-** содержат накопленную по всем тарифам активную энергию (A) и соответственно реактивную энергию (R) прямого/обратно (+/-) направления. Все группы **Энергия** имеют одинаковый набор выходов, хранящие накопленное значение энергии соответствующей категории группы, за различные периоды времени – от последнего сброса, за текущие и предыдущие сутки, за текущий и предыдущий год, а также за прошедшие месяцы.

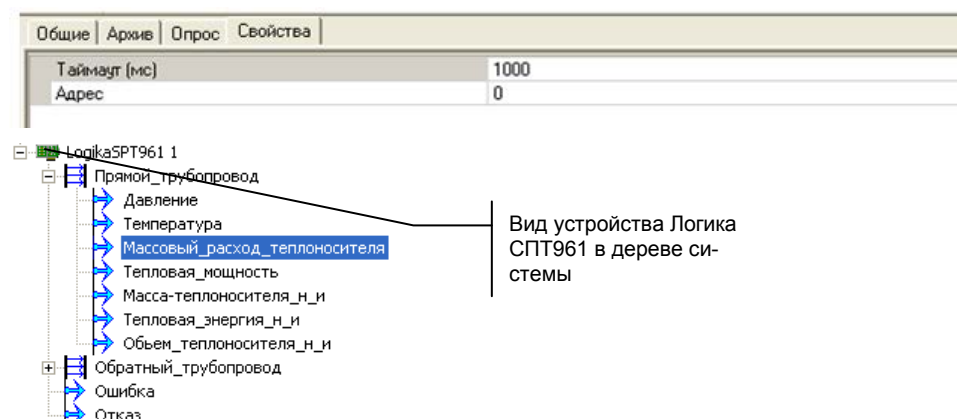


Группа **Текущая запись профиля мощности** содержит в себе архивные значения, считываемые из счетчика.



Группа **Журналы**. Меркурий 230 хранит в памяти журнал событий – пропадание фаз, отключение питания, открытие крышки и т.д. Журнал представляет собой кольцевой буфер на 10 записей. Более подробное описание данного функционального блока можно прочитать вызвав контекстное меню **Справка** в среде MasterSCADA.

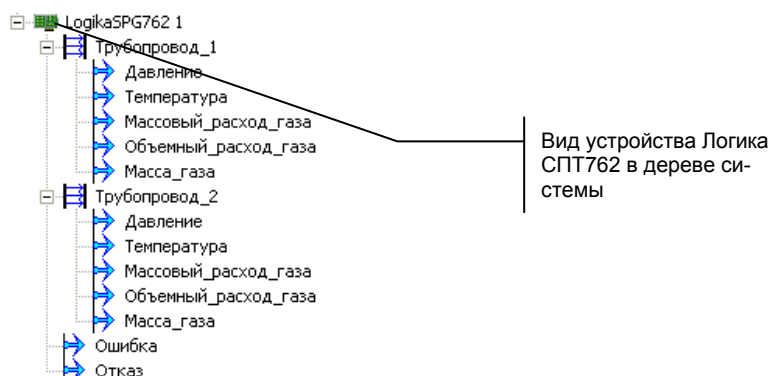
Настройка протокола тепловычислителя **Логика СПТ961** для опроса по интерфейсу RS-485/232 осуществляется на странице Свойств, где настраивается адрес устройства и Таймаут позволяющий определить период времени, по истечению которого будет формироваться сигнал ошибки с последующим всплывающим сообщением.



Прибор содержит две группы входов имеющих одинаковые наборы входов: **Прямой трубопровод** и **Обратный трубопровод**. Используются следующие параметры:

- **Давление** – измеренное значение давления в трубопроводе;
- **Температура** – измеренное значение температуры в трубопроводе;
- **Массовый расход теплоносителя** – измеренное значение температуры в трубопроводе;
- **Масса теплоносителя_н_и** – вычисленное значение массы теплоносителя (нарастающий итог);
- **Тепловая энергия_н_и** – вычисленное значение тепловой энергии (нарастающий итог);
- **Объем теплоносителя_н_и** – вычисленное значение объема теплоносителя (нарастающий итог);
- **Отказ** – на выходе формируется ВКЛ в случае отсутствия связи с прибором;
- **Ошибка** – на выходе формируется статус ошибок прибора. Биты состояния: 0 – отказ прибора, 6 – ошибка по контрольной сумме, 7 – ошибка данных в принятом буфере.

Настройка корректора **Логика СПТ762** происходит аналогично настройке СПТ961 описанной выше. Корректор предназначен для измерения электрических сигналов, соответствующих параметрам технических газов различного состава, и последующего вычисления расхода и объема газа, приведенных к стандартным условиям.



Прибор содержит две группы - **Трубопровод_1** и **Трубопровод_2**, которые имеют одинаковые наборы входов:

- **Давление** – измеренное давление газа в трубопроводе;
- **Температура** – измеренная температура газа в трубопроводе;
- **Массовый расход газа** – измеренное значение массового расхода газа;
- **Объемный расход газа** – вычисленное значение объемного расхода газа;
- **Масса газа** – масса газа нарастающим итогом;
- **Ошибка** – всегда ноль;
- **Отказ** - на выходе формируется ВКЛ в случае отсутствия связи с прибором.

Настройка тепловычислителя **Логика СПТ961** аналогична другим подобным устройствам, за исключением настройки свойства **Смещение времени**, которое позволяет задать смещение для задания часового пояса. Тепловычислитель предназначен для учета тепловой энергии и теплоносителя в закрытых и открытых водяных системах теплоснабжения, рассчитан на два тепловых ввода, на каждом из которых могут быть установлены три датчика объема, два датчика температуры и два датчика давления.

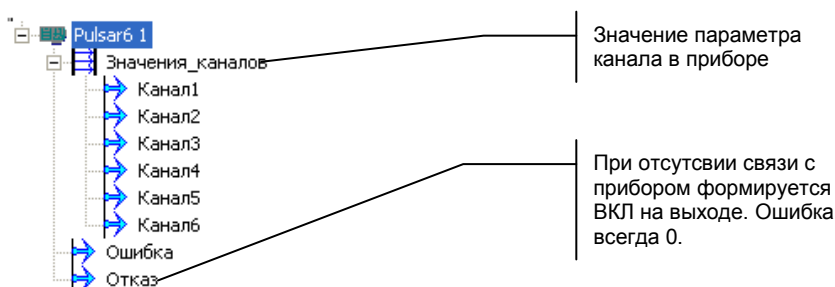
Общие Архив Опрос Свойства	
Таймаут (мс)	1000
Адрес	0
Смещение_времени	3



Описание выходов функционального блока:

- **Схема потребления.** Значения на выходе могут быть от 0 до 10. Выход передает установленную схему потребления по каждому тепловому вводу (подробнее в документации к прибору);
- **Текущий_расход_по_трубопроводу_*n*** – вычисленное от датчика объема текущее значение расхода;
- **Текущее_давление_в_трубопроводе_*n*** – значение давления измеренное датчиком;
- **Текущая_температура_в_трубопроводе_*n*** – значение температуры измеренное датчиком;
- **Тотальный_объем_в_трубопроводе_*n*** – нарастающий итог объема теплоносителя за время счета;
- **Тотальная_масса_в_трубопроводе_*n*** – нарастающий итог массы теплоносителя за время счета;
- **Тотальная_тепловая_энергия_по_вводу** – нарастающий итог вычисленной тепловой энергии по вводу;
- **Суммарное_время_счета** – установленное время счета;
- В группе **ТВ*n*_Часовой_архив** находится архив значений прибора. У архивируемых параметров на закладке Архив по умолчанию стоит галочка Архивировать;
- **Отказ** – на выходе формируется ВКЛ в случае отсутствия связи с прибором;
- **Ошибка** – аналогично Логика СПТ961.

При включении протокола **Пульсар** в COM порт можно добавить два устройства – **Пульсар 6** и **Пульсар 10**. Пульсар 6 - прибор с шестью счетными каналами и жидкокристаллическим дисплеем. Пульсар 10 – прибор с десятью счетными каналами без элементов индикации. Приборы предназначены для коммерческого и технологического учета воды, газа, электроэнергии. Вычисление идет нарастающим итогом по каждому каналу. Свойства Пульсара аналогичны устройству Логика СПТ 762. Вид устройства в дереве системы приведен ниже.



Модули ввода-вывода (платы)

Модули ввода-вывода, в данном случае, это платы, которые подсоединяются непосредственно к материнской плате компьютера.

Элементы дерева объектов

Дерево Объектов предназначено для размещения в нем **Объектов**, которые отвечают за порядок вычислений, выбор компьютера, содержат Функциональные блоки, Переменные и их Группы, управляют доступом к внешним данным.

Начальным элементом Дерева объектов является **Главный Объект**, в котором пользователь размещает остальные Объекты согласно желаемой конфигурации.

Главный Объект может быть только один. Он может содержать только другие Объекты, которые автоматически наследуют его свойства (нельзя разместить в нем переменные, функциональные блоки). Сам же Главный Объект наследует свойства назначенного компьютера (или Системы, если компьютер не назначен).

Существуют следующие основные виды элементов **Дерева Объектов**:

- Объекты;
- Функциональные блоки;
- Группы переменных;
- Переменные.

Объекты

Объект - это элемент программы, предназначенный для размещения в нем всех других элементов - переменных и функциональных блоков, а также других объектов. Объект выполняет: организацию пересчета функциональных блоков, прием и передачу данных внешним переменным, порядок вычислений.

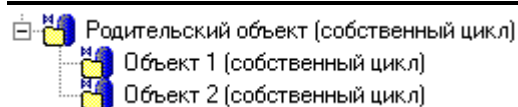
Объекты имеют набор документов:

- **Окна** - стандартные документы MasterSCADA (мнемосхемы и т д), количество, названия, размеры которых устанавливаются на странице **Окна** для **Системы** (и могут быть изменены для каждого Компьютера);
- Тренды;
- Рапорты;
- Архивы;
- Журналы;
- **Другие документы** (любые word и подобные файлы).

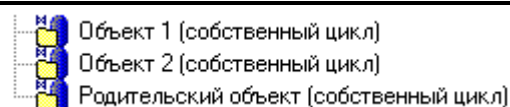
Все объекты с точки зрения организации порядка вычислений подразделяются на два типа: имеющие собственный цикл вычислений и не имеющие собственного цикла вычислений. Объекты первого уровня (помещенные в главный) всегда имеют собственный цикл.

Объекты, **имеющие собственный цикл** вычисляются независимо от родительского объекта.

Первый вариант размещения объектов



Второй вариант размещения объектов

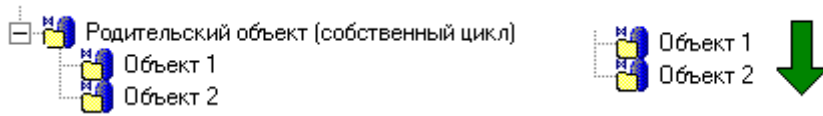


Порядок вычислений в обоих случаях один и тот же - все три объекта имеют разные циклы вычислений, которые работают параллельно и независимо. Порядок следования объектов (имеющих собственный цикл) в дереве не имеет значения.

Объекты, **не имеющие собственного цикла вычислений** вычисляются последовательно сверху вниз в цикле своего родительского объекта.

Первый вариант размещения объектов

Второй вариант размещения объектов

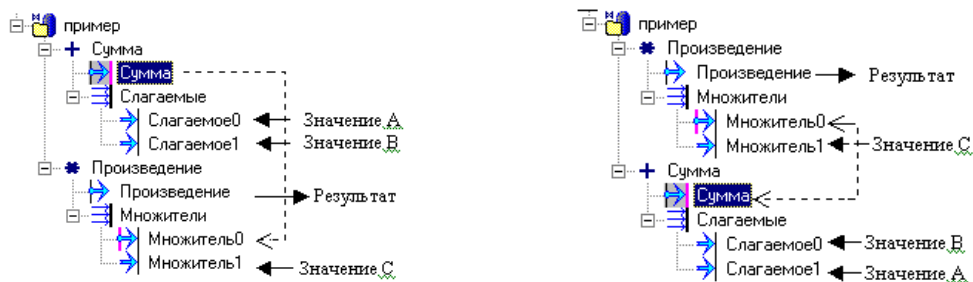


В этом случае гарантирован порядок вычислений последовательно сверху вниз для объектов и функциональных блоков, расположенных на одном уровне иерархии. При разработке объекта необходимо учитывать этот порядок, особое внимание обращая на связи с вышестоящими элементами.

- **Пример:** представим, что необходимо произвести вычисления по формуле $(A + B) \times C$.

Правильный вариант размещения объектов

Не правильный вариант размещения объектов



На первый взгляд результат в обоих случаях один и тот же: значения A и B - слагаемые, сумма - первый множитель, значение C - второй множитель. Если бы это были константы, то все бы так и было. Но если значения A и B - обновляются с каждым опросом (циклом), получается следующее: сначала вычисляется произведение. Считываются данные входа Множитель0 и Множитель1, но Сумма еще не вычислялась и на входе Множитель0 находится значение, вычисленное Суммой за предыдущий цикл. Поэтому, чтобы получить корректную работу формулы, необходимо чтобы Сумма стояла выше, чем Произведение.

Балансировка производительности - автоматическое распределение программой ресурсов компьютера между объектами. Включение этого режима обеспечивает выравнивание отношения реального времени опроса к заданному времени опроса у объектов компьютера. Настройка Балансировки производительности по умолчанию для компьютеров системы осуществляется на странице свойств Системы **Опрос**. В соответствии с идеологией построения проекта сверху вниз, настройки системы наследуются компьютерами при вставке их в проект, но в любой момент для каждого конкретного компьютера могут быть переопределены.

Минимальный период балансировки - это минимальный период цикла, который будет установлен всем объектам, имеющим меньший, чем этот. Необходим для случая, если у какого-либо объекта установлен не разумный период опроса.

Приоритет объекта - отношение времени, затрачиваемого процессором для обработки данного объекта к времени, затрачиваемого на обработку всех объектов компьютера. Для предотвращения ситуации когда некорректная работа какого-либо Объекта может негативно отразиться на работе системы, введено ограничение максимального приоритета.

- **Пример:** Пусть в проекте есть два объекта, относящихся к одному компьютеру, у которого не достаточно ресурсов чтобы успевать пересчитывать объекты за заданные периоды. Тогда включение балансировки повлияет на работу компьютера приблизительно следующим образом:

	Объект 1	Объект 2
Заданный период опроса	100	50
При выключенной балансировке	150 / 50%	150 / 50%
Реальный период опроса / приоритет объекта		

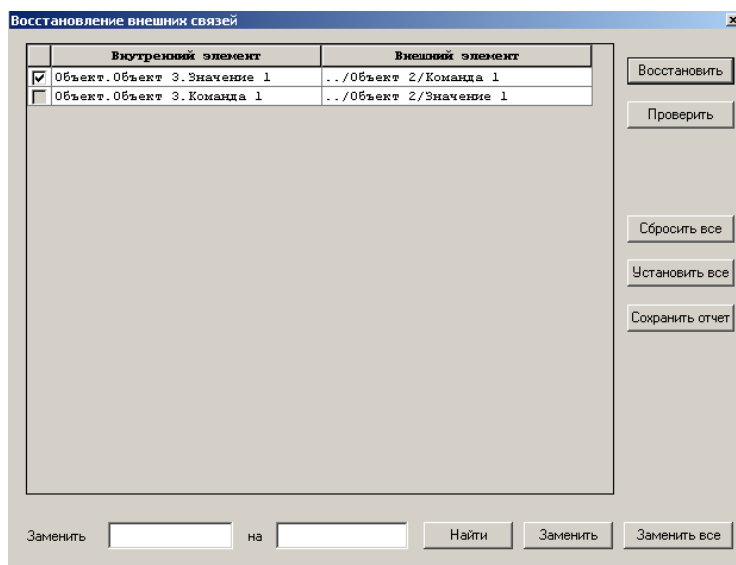
При включенной балансировке
Реальный период опроса / приоритет объекта

200 / 33% 100 / 67%

При **тиражировании Объектов** сохраняются все внутренние связи, настройки, документы и настройки документов. Есть три способа тиражирования объектов:

- Копирование;
- Дублирование;
- Помещение в библиотеку и вставка из библиотеки.

При тиражировании Объектов, имеющих внешние связи эти связи восстанавливаются, если такое возможно. При отсутствии возможности восстановления, есть возможность редактирования адресов не восстановившихся связей. Установка связи производится с тем объектом (переменной), который в дереве объектов расположен в том же месте, где был связанный объект (переменная) у оригинала в момент копирования (помещения в Библиотеку). При вставке, если элемент имеет внешние связи, появится диалог приблизительно такого вида.



В диалоге следует отметить флажками связи, которые действительно требуется восстановить.

Кнопка	Назначение
Восстановить	Восстанавливаются все отмеченные связи
Сбросить все	Все связи в таблице помечаются как не требующие восстановления (флаги напротив адресов снимаются).
Установить все	Все связи, которые могут быть установлены, помечаются как требующие восстановления (при этом напротив каждой связи устанавливается флаг)
Сохранить отчет	Сохраняется отчет для списка не восстановленных связей в текстовом файле.
Проверить	Проверяется корректность отредактированной связи. Если связь восстановить можно, то флаг, показывающий возможность восстановления связи устанавливается.
Найти	При нажатии на эту кнопку происходит проверка существования не восстановленной связи, внешний элемент которой имеет имя введенное в поле Заменить. Если такая связь существует, то она выделяется.
Заменить	Для выделенной связи в списке позволяет заменить имя внешнего элемента, заданного в поле Заменить, на новое (вводится в поле <i>На</i>)
Заменить все	В списке не восстановленных связей ищутся связи, внешний элемент которых имеет имя, введенное в поле Заменить. Если такие связи есть, то имя внешнего элемента у них заменяется на заданное в поле <i>На</i> .

При редактировании адресов связей можно использовать Внешние элементы, обозначающий переход к объекту верхнего уровня.

- ❖ **Внимание!** При групповом выделении и копировании сразу нескольких элементов, (выделенных на одном уровне), их связи будут считаться внешними и устанавливаться по правилам внешних связей, как если бы их копировали по одному!

В MasterSCADA имеются папки для хранения объектов, созданных пользователем - **Библиотеки**. Помещенные в Библиотеку объекты сохраняют все свои свойства, в том числе внутренние связи, внешние связи, подлежащие восстановлению, мнемосхемы, другие документы и т. п. Поместить объект в Библиотеку можно через контекстное меню, всплывающее при нажатии правой клавиши мыши на объекте, либо воспользоваться командой **Поме-**

стить в библиотеку в меню Правка. Удаление объекта из Библиотеки осуществляется через контекстное меню командой Удалить.

- ❖ **Внимание!** Удаленный из Библиотеки объект нельзя вернуть обратно.

В то же время большое число объектов в Библиотеке никак не влияет на производительность системы, и не требует заметных ресурсов. Расчет и Событие также можно помещать в библиотеку. При помещении в Библиотеку они также сохраняют свои настройки и свойства. Таким образом, использование библиотечных объектов является мощнейшим средством оптимизации процесса разработки проектов так как значительным образом сокращает время, потраченное на создание новых объектов. Кроме того, это позволяет создавать один и тот же проект несколькими разработчиками параллельно. Для создания Библиотеки используйте команду **Создать библиотеку** из меню **Сервис**. После этого ее имя появится в списке Библиотек в палитре и будет отображаться в контекстном меню объекта в дереве в команде Поместить в библиотеку. Осталось только наполнить Библиотеку полезными объектами.

Переменные

В MasterSCADA есть следующие виды **Переменных**:

В дереве Системы:

- **ОПС-переменные** (для чтения, для записи, для чтения и записи)

В дереве Объектов:

- **Значение**;
- **Команда**;
- **Расчет**;
- **Событие**;
- **Вход функционального блока**;
- **Выход функционального блока**;

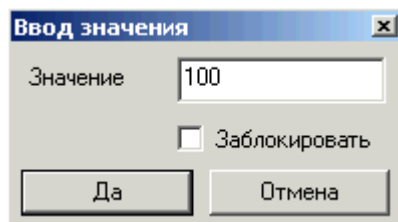
Переменные выполняют следующие функции:

- **Запись (прием)** данных в переменную. Переменная принимает данные извне и помещает их в собственное хранилище;
- **Хранение данных**. То есть данные содержатся в самой Переменной до тех пор пока не будут перезаписаны;
- **Передача текущих данных** по всем установленным связям (кроме переменных-входов). Это действие совершается сразу после выполнении действия данных, то есть, как только данные приняты (записаны), они сразу передаются по связям.

Значение используется для передачи данных. Значение отображается в дереве объектов значком ➔


В Значение данные могут поступать одним из следующих способов:

- От ОПС переменной, Выхода ФБ, Расчета, Команды;
- По расписанию Объекта или по действию События;
- Со страницы свойств Значения Данные. Кроме того, в режиме исполнения по щелчку на данных Значения в дереве объектов, открывается диалоговое окно, также позволяющее установить значение на выходе переменной:



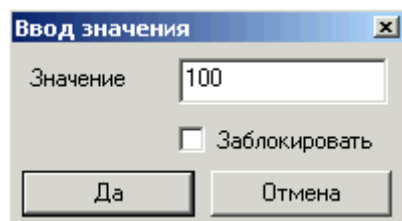
Способы использования данных Значения:

- Передавать на Вход документа;
- Перетаскать в окно документа (для наблюдения). Для получения более подробной информации;
- Использовать в качестве источника данных в формуле, тренде, рапорте.

Команда используется для передачи данных. Ее основное применение - быть получателем данных из мнемосхемы (или другого стандартного документа). Команда отображается в дереве объектов значком .


В Команду данные могут поступать одним из следующих способов:

- От выхода документа (можно установить связь с несколькими выходами документов, данные будут поступать от того, кто последним изменил значение);
- По расписанию Объекта или по действию События;
- От связанного с Командой элемента мнемосхемы. Такой элемент создается перетаскиванием Команды из дерева Объектов в мнемосхему;
- Со страницы свойств Команды Данные. Кроме того, в режиме исполнения по щелчку на данных Команды в дереве объектов, открывается диалоговое окно, также позволяющее установить значение на выходе Команды.



Способы использования Команды:

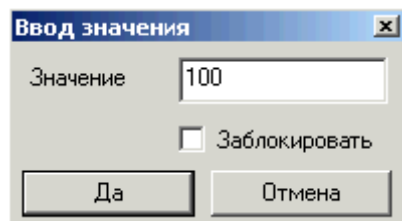
- Создать связь с OPC переменной, Входом ФБ или Значением;
- Перетащить в окно документа;
- Использовать в качестве источника данных в формуле, тренде, рапорте.

Расчет используется для передачи данных. Главной его особенностью является наличие формулы, по которой производится вычисление значения расчета. В формуле можно использовать в качестве переменной сам **Расчет**, в этом случае будет подставляться предыдущее вычисленное значение. Расчет отображается в дереве объектов значком . Внутри расчета все выражения приводятся к одному из следующих типов данных: логический, целый или вещественный.

В Расчет данные могут поступать одним из следующих способов:


- В результате вычислений по формуле;
- По расписанию Объекта или по действию События;
- Со страницы свойств Расчета Данные.

Кроме того, в режиме исполнения по щелчку на данных Расчета в дереве объектов, открывается диалоговое окно, также позволяющее установить значение на выходе Расчета:



Способы использования данных Расчета:

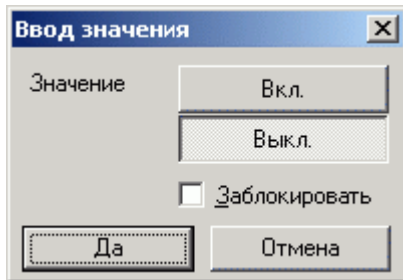
- Создать связь с OPC переменной, Входом ФБ или Значением для передачи в них данных;
- Перетащить в окно документа для наблюдения данных;
- Использовать в качестве источника данных в формуле (другого Расчета или События), тренде, рапорте.

Событие в MasterSCADA - это однократная информация о переходе результата вычислений формулы из Ложь в Истина. События используются для выдачи сообщений и выполнения действий. Событие отображается в дереве объектов значком . Оно всегда имеет дискретный тип данных.

Если значение События переходит из Ложь в Истина, то выдается сообщение (если оно создано) или выполняются действия (если они назначены), которые настраиваются на соответствующих страницах событий.

В Событие данные могут поступать одним из следующих способов:

- В результате вычислений по формуле;
- По расписанию Объекта или по действию другого События;
- Со страницы свойств События Данные. Кроме того, в режиме исполнения по щелчку на данных События в дереве объектов, открывается диалоговое окно, также позволяющее установить значение на выходе События.





Способы использования данных События:

- Создать связь с ОРС переменной, Входом ФБ или Значением;
- Перетащить в окно документа (для наблюдения);
- Использовать в качестве источника данных в формуле (расчета или другого События), тренде, рапорте.



Расчет и событие могут быть помещены в **Библиотеку** для последующего использования при создании других проектов. При помещении в Библиотеку сохраняется формула Расчета, а также все его свойства и настройки.

Вход функционального блока - это элемент функционального блока, на который возложена реализация следующих трех функций:

- Запись (прием) данных во Вход. Вход принимает данные извне и помещает их в собственное хранилище. Эта функция вызывается передачей данных выходом. Кроме того, Вход информирует другие элементы о приеме данных. Это используется либо на странице свойств, либо если функциональный блок использует метод опроса **по изменению входов**, то блоку будет установлен специальный признак, что он требует пересчета;
- Хранение данных. То есть данные содержатся в самом Входе до тех пор пока не будут перезаписаны;
- Чтение данных из Входа. Данные читаются из Входа функциональным блоком и страницей свойств.

Входы функционального блока в дереве объектов отмечаются значком , который приобретает вид , если Вход имеет связь. Вход может иметь только одну связь с каким-либо выходом.

Выход функционального блока - это элемент функционального блока, который выполняет следующие функции:

- Запись (прием) данных в Выход. Данные поступают из функционального блока или из страницы свойств в режиме исполнения. Это действие вызывается пересчетом функционального блока;
- Хранение данных. Данные хранятся в самом Выходе до тех пор, пока не будут перезаписаны;
- Установление и разрыв связей с Входами. Выход может иметь связь со сколь угодно большим числом Входов. Выходы, не имеющие связей отображаются в дереве значком . Для установления связи пользователю достаточно выделить Выход и перетащить его мышью к нужному Входу. Если установлена хотя бы одна связь, Выход приобретает вид ;
- Передача текущих данных по всем установленным связям. Это действие вызывается при выполнении действия Запись данных в Выход, то есть, как только данные приняты (записаны), они сразу передаются по связям. В свою очередь, выполнение пере-

дачи данных повлечет запись данных во Вход функционального блока, имеющего связь с Выходом.

Данные Выхода ФБ можно использовать следующим образом:

- Создать связь с ОПС переменной, Входом ФБ или Значением;
- Перетащить в окно документа (чтобы наблюдать);
- Использовать в качестве источника данных в формуле, тренде, рапорте.

Для установления связи между входом и выходом необходимо перетащить мышью одну переменную на другую. Приведенная ниже таблица включает в себя все возможные комбинации. В столбиках таблицы находятся переменные исполнительной системы, в строках - все элементы, с которыми создаются связи. Связи, которые возможно установить отмечены плюсами.

	ОПС переменная для чтения	ОПС переменная для записи	ОПС переменная для чтения и записи	Команда	Расчет	Значение	Событие	Вход ФБ	Выход ФБ
ОПС переменная для чтения		+	+			+		+	
ОПС переменная для записи	+		+	+	+		+		+
ОПС переменная для чтения и записи	+	+	+	+	+	+	+	+	+
Команда		+	+			+		+	
Расчет		+	+			+		+	
Событие		+	+			+		+	
Значение	+		+						
Вход ФБ	+		+	+	+		+		+
Выход ФБ		+	+			+		+	
Формула Расчета (События)	+		+	+	+	+	+		+
Окно мнемосхемы	+		+	+	+	+	+		+
Вход мнемосхемы	+		+	+	+	+	+		+
Выход мнемосхемы		+	+	+					
Тренд	+		+	+	+	+	+		+
Ячейка рапорта	+	+	+	+	+	+	+	+	+

- ❖ **Важно!** Параметры формулы расчета или события могут быть связаны только с переменными, принадлежащими тому же компьютеру, что и сам расчет/событие. В противном случае, данные от переменных в расчет/событие передаваться не будут.

Каждой переменной объекта в MasterSCADA (кроме входов функциональных блоков) можно присвоить **Диапазон значений**. Он выбирается из выпадающего списка на закладке **Общие** у переменной.

Существуют диапазоны следующих типов:

- Аналоговые - применимы только для аналоговых переменных;
- Дискретные - для дискретных переменных;
- Перечислимые - могут быть использованы только для переменных, имеющих перечислимый тип;
- Интервальные - только для переменных интервального типа.

Настройка диапазонов производится на странице **Шкалы** у Системы. Аналоговые диапазоны имеют атрибуты указанные в таблице ниже.

Атрибут	Описание
Имя интервала	Отображается в выпадающих списках выбора аналогового диапазона.
Начало	Число, от которого вычисляются нижние аварийная и предупредительная границы.
Конец	Число, от которого вычисляются верхние аварийная и предупредительная границы.
Точность	Сколько знаков после запятой отображать.
Единица измерения	Отображается после значения переменной везде где выводится это значение.

- ❖ **Важно!** *Данные атрибуты передаются вместе с переменной, например в трендах начало и конец определяют минимум и максимум оси значений, в элементе управления Индикатор соответствующим образом раскрашивается шкала, единица измерения пишется в пользовательских сообщениях и т. п.*

По атрибутам Начало и Конец вычисляются следующие величины:

- Нижняя аварийная граница;
- Нижняя предупредительная граница;
- Верхняя аварийная граница;
- Верхняя предупредительная граница.

Кроме того, можно задать еще **максимальную скорость изменения** и **мертвую зону** для контроля значений. Эти величины будут установлены автоматически переменной объекта при выборе для нее данного интервала. Их можно изменять для каждой конкретной переменной, т.е. они не являются жестко заданными, просто устанавливаются по умолчанию.

Когда значение переменной переходит какую-либо границу (или скорость изменения превышает максимальную) автоматически происходит следующее:

- Выводимое значение переменной окрашивается в соответствующий выбранный цвет (или начинает мигать);
- Формируется соответствующее системное сообщение.

Дискретные интервалы имеют только один атрибут - пару названий для возможных значений переменной (для **Истина** и для **Ложь**). Эти названия будут отображаться вместо **Вкл** и **Выкл**, которые установлены по умолчанию. В перечислении (перечислимые переменные) определяется набор пар целочисленное значение - строка. Перечисления удобно использовать, если переменная может принимать только значения, заданные в наборе. Интервальные диапазоны выполняют ту же функцию, что и перечислимые, за исключением того, что состояние переменной определяется не целым числом, а интервалом.

Переменные MasterSCADA (значение, команда, событие, расчет, выходы функциональных блоков и OPC переменные для чтения) могут быть использованы как визуальные функциональные блоки. Их можно перетаскивать в мнемосхемы, для того чтобы просматривать значения переменных или устанавливать данные на их выходах. При перетаскивании переменных в мнемосхему Вы можете использовать как левую, так и правую кнопки мыши:

- По левой кнопке в мнемосхему вставляется элемент управления по умолчанию;
- По правой кнопке открывается контекстное меню, позволяющее выбрать тип элемента управления.

Для просмотра и установления данных на выходах переменных возможно использование и традиционного способа, заключающегося в создании элемента управления в мнемосхеме и динамизации свойств этого элемента от входов/выходов мнемосхемы, но более предпочтительным для данной цели является именно перетаскивание, т.к. у этого способа существует следующий ряд преимуществ:

- Быстрота создания в мнемосхеме связанного с переменной элемента управления: для создания связи не нужно заводить входы/выходы мнемосхемы и динамизировать свойства элементов - достаточно просто перетащить переменную в мнемосхему и настроить внешний вид элемента управления;
- Автоматическое наследование элементом управления диапазона шкал, единиц измерения, формата значения и аварийных зон переменной.

В зависимости от типа переменной возможны следующие элементы управления:

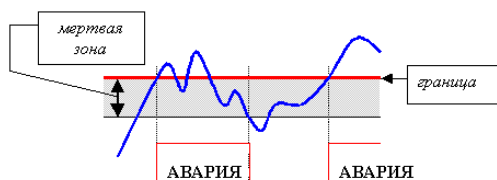
- Редалируемое значение;
- Задатчик;
- Значение;
- Стрелочный прибор;
- График;
- Индикатор;
- Индикатор графический;
- Индикатор мнемонический;
- Зона выбора.

Более подробно описание элементов управления доступных для каждого из типов переменных описано в справке программы.

Поле на странице свойств переменной **Период опроса** - это наименьший промежуток времени, через который производится опрос переменных. Существует два основных вида опроса переменных:

- Периодический (через каждые Период опроса миллисекунд);
- По изменению входов (как только значение, подаваемое на вход изменится более, чем на значение Мертвой зоны).

Поле на странице свойств переменной **мертвая зона** - это минимальная разность между текущим и последним зафиксированным значениями входа, которая приводит к опросу функционального блока. В случае, если изменение значения на входе инициирует пересчет функционального блока, текущее значение входа фиксируется и используется для сравнения с последующими. В установках параметров опроса Мертвая зона - это максимальное отклонение сигнала от установленного значения, при котором сигнал считается не изменяющимся. Мертвая зона служит для предотвращения дребезга сигнала (например, аварийного) при переходе параметром границы.



Как видно из рисунка, если параметр возвращается из аварийного состояния в нормальное на малое значение (меньше мертвой зоны), аварийный сигнал снят не будет.

В программе допускается устанавливать связь между переменными разных типов. При этом, если в результате этой связи происходит потеря точности, выдается соответствующее предупреждение. При установлении связей разных типов необходимо, чтобы значение выхода уместилось в диапазон входа, в обратном случае данные не будут передаваться. Ограничения на значения преобразования данных при связи переменных разных типов более подробно описаны в справке программы.

В MasterSCADA существуют следующие признаки качества переменных:

Название	Описание
Ошибка конфигурации	Алгоритм имеет проблемы с конфигурацией, например, не все связи или настройки заданы.
Нет соединения	Алгоритм имеет проблемы со связью, например, у входа алгоритма оборвана связь с переменной алгоритма-источника данных.
Ошибка устройства	Предписанные алгоритмом операции нельзя выполнить (например, деление на нуль или переполнение порядка).
Ошибка датчика	Хотя бы одна входная переменная не удовлетворяет условиям, накладываемым на входные переменные алгоритмом, или хотя бы одна входная переменная пришла с любым из плохих признаков качества.

Название	Описание
Последнее известное значение	Сбой связи. Последнее опрошенное значение доступно. Время опроса имеется в отметке времени, сопровождающей значение.
Нет связи	Связи с источником данных для какого-либо из входов нет. Последнее опрошенное значение не существует. Или алгоритм работает, но выходное значение пока еще не сформировано (например, расчет интеграла).
Не обслуживается	При существующих настройках алгоритма данный выход не формируется.

С помощью **Зоны выбора** в мнемосхеме можно создавать активные области, позволяющие в режиме исполнения отслеживать щелчок левой кнопки мыши внутри ее границ, и производить при этом определенные действия. Создать зону выбора можно перетаскив Событие в мнемосхему правой кнопкой мыши, затем в открывающемся контекстном меню выбрать **Зона выбора**. При активизации зоны выбора (щелчке левой кнопкой мыши в режиме исполнения внутри области, ограничиваемой зоной выбора) происходит следующее:

- На выходе События либо выдается импульс, либо устанавливается сигнал Истина, который снимается при повторной активизации зоны выбора. По умолчанию на выходе События выдается импульс. Настройка, определяющая фиксировать сигнал на выходе События или нет, задается в мнемосхеме (свойство зоны выбора Фиксировать значение);
- При переходе выхода связанного с зоной выбора События из Ложь в Истина, формируется сообщение, а также выполняются все действия, заданные для этого события.

Для того, чтобы в режиме исполнения зона выбора корректно обрабатывала события мыши, необходимо, чтобы на мнемосхеме она находилась поверх остальных элементов.

Группы переменных

Это элементы, сделанные для того, чтобы ради удобства помещать в них переменные. Можно различить группы входов-выходов функциональных блоков и группы переменных объектов. Поместить в Группу переменных объектов новую переменную возможно как через контекстное меню, всплывающее при нажатии правой клавиши мыши, так и с помощью команд меню Добавить. В группы входов-выходов функциональных блоков таким образом переменные не помещаются. Изменение числа входов-выходов в этом случае производится со страниц свойств функционального блока, и только в том случае, если такое изменение предусмотрено.

Функциональные блоки

Функциональные блоки - это основные элементы программы, отвечающие за управление данными. Они предназначены для выполнения следующих двух функций:

- Содержать в себе Входы и Выходы. Некоторые Функциональные блоки содержат также группы входов и выходов;
- Предоставлять функцию, вызов которой приводит к преобразованию данных на Выходах в зависимости от данных на Входах (пересчет функционального блока).

Визуальные функциональные блоки

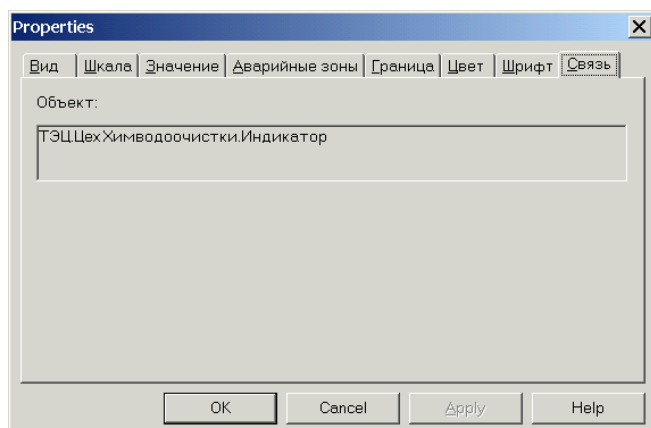
Визуальный функциональный блок (ВФБ) совмещает в себе:

- Обычный функциональный блок;
- Графический элемент мнемосхемы (элемент управления ActiveX).

Использование ВФБ может значительным образом облегчить создание мнемосхемы объекта. Для размещения в мнемосхеме достаточно перетаскать блок мышью в нужное место в окне редактора мнемосхем. Внутренние связи между ФБ и его графическим элементом устанавливаются автоматически. Одному функциональному блоку может соответствовать любое число элементов в различных мнемосхемах. Кроме того, переменные MasterSCADA также могут являться ВФБ.

ВФБ имеет готовое изображение и свойства в мнемосхеме (как графический элемент управления), и уже установленные связи с функциональным блоком в дереве объектов. Это означает, что после перетаскивания функционального блока больше ничего настраивать не надо, поведение его изображения в режиме исполнения будет зависеть от состояния входов в дереве объектов и настроек ВФБ. И наоборот, работая с элементом управления в мнемосхеме,

схеме, оператор может управлять выходами и состоянием ВФБ. Связи между ВФБ и его графическим элементом не отображаются в палитре MasterGraph. На закладке Связь каждого графического элемента ВФБ отображается его связь с самим ВФБ в проекте и путь до этого ВФБ в дереве проекта.



При удалении ФБ из дерева объектов связь элемента с ним разрывается. На закладке Связь появится надпись **Связь не определена**, а сам элемент будет отображаться в виде **X**. Чтобы восстановить связь, следует перетащить необходимый ВФБ из дерева проекта на закладку свойства выбранного элемента в палитре мнемосхемы, а затем убедиться, что на закладке связь появилась соответствующая запись. В обоих случаях изменение изображения графического элемента (на крест или обратно) произойдет только после выполнения команды **Обновить связи** меню **Сервис**.

Страницы свойств элементов

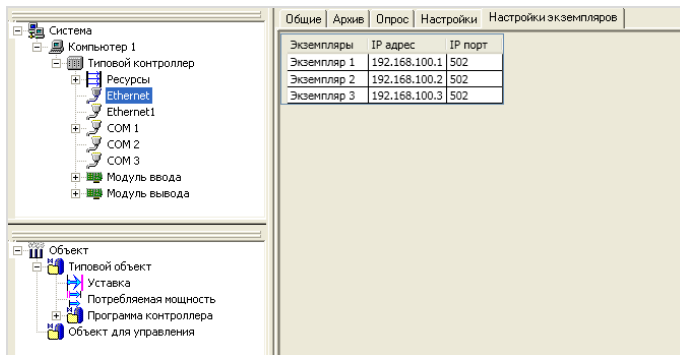
В то время как структура проекта определяется с помощью дерева системы и дерева объектов, настройка каждого элемента этих деревьев начиная с самого основного - **Системы** и заканчивая каждой переменной производится на страницах свойств. Набор страниц свойств и их вид для каждого элемента индивидуальны. Так как каждый элемент может быть настроен индивидуально, настройка получается очень много. Поэтому общие свойства всех элементов автоматически устанавливаются как у родительского элемента, т.е. **наследуются**. В каждый момент времени открыты страницы свойств того элемента, который выделен в дереве системы или дереве объектов.

Типизация

Инструмент **Типизация** предназначен для создания проектов содержащих несколько полностью идентичных объектов. Этим он отличается от инструмента **Шаблон**, использование которого позволяет создавать объекты, имеющие отличия от исходного шаблонного объекта.

При использовании этого инструмента создается типовой объект и типовой контроллер, а затем создается список экземпляров – с именами и привязками к контроллеру. В проекте фигурирует единственный объект, даже, если он имеет сотни экземпляров. Благодаря этому, дерево проекта не только проще для обзорности и понимания, но и гораздо быстрее открывается при первоначальной загрузке проекта или закрывается при его закрытии.

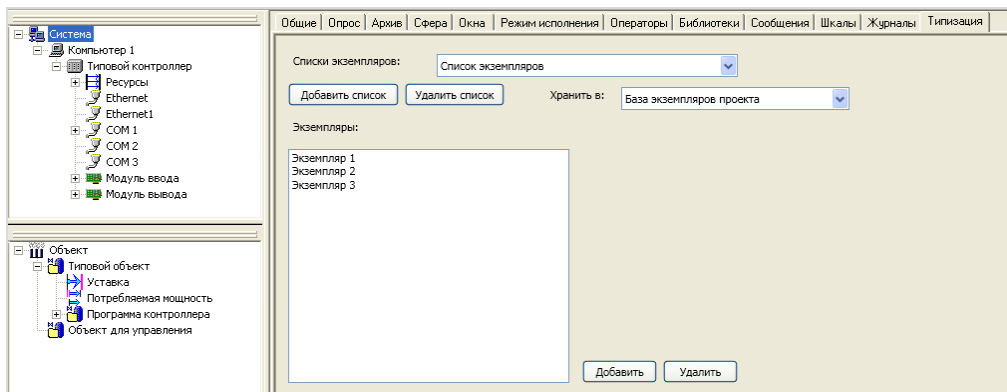
Например, мы имеем объект электросчетчик. Поскольку тип счетчика у нас один, а на автоматизируемом объекте их установлено много, инструмент типизация позволит создать в проекте единственный счетчик со всеми его алгоритмическими обработками и документами, а также задать источники данных для каждого экземпляра этого объекта. Создание списка экземпляров осуществляется у системы, на закладке **Типизация**. На этой странице необходимо добавить список (или несколько списков) с именами экземпляров. После этого созданный список необходимо применить к созданному типовому объекту. Присваивание осуществляется на закладке **Типизация объекта**. На этой же странице осуществляется настройка опроса, переключения, и установки текущего экземпляра. Аналогичным образом происходит присваивание списка типизированному контроллеру. Для настройки параметров контроллеров, используется закладка **Настройка экземпляров порта Ethernet**. На данной странице необходимо задать IP адреса контроллеров и порты связи.



После этого каждый экземпляр объекта будет исполняться независимо. На мнемосхеме, в свойствах кнопки, типового объекта можно установить принадлежность кнопки конкретному экземпляру списка. В этом случае, нажатие на кнопку будет открывать документ данного экземпляра.

На мнемосхеме также можно расположить контрол **Список экземпляров** (закладка окна палитры мнемосхемы). С помощью данного контрола можно в режиме исполнения управлять списком экземпляров - добавлять, удалять, дублировать, редактировать. Привязка к конкретному списку осуществляется в панели свойств контрола.

Разработчик может ограничить количество создаваемых экземпляров, для этого нужно использовать **ФБ Количество экземпляров** из закладки служебные. Данный ФБ подсчитывает количество своих копий, и если количество копий превышает заданное в настройках, его выход переходит в состояние Ложь. Используя данное событие можно блокировать отдельные функции, сформировать сообщение, или остановить режим исполнения. На закладке Список экземпляров производится создание списков экземпляров объектов.



Существуют следующие Элементы управления списками экземпляров.

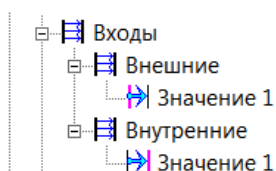
Элемент	Описание
Поле Списки экземпляров	Содержит все созданные в проекте списки экземпляров.
Кнопка Добавить список	Добавляет экземпляр в список.
Кнопка Удалить список	Удаляет экземпляр из списка.
Поле Хранить в:	Позволяет выбрать способ хранения данных экземпляров.
Поле Экземпляры	Все добавленные экземпляры в текущий список.
Кнопка Добавить	Добавляет экземпляр в текущий список.
Кнопка Удалить	Удаляет текущий экземпляр текущего списка.
Кнопка Переименовать	Переименовывает экземпляр в текущем списке.
Кнопка Дублировать	Дублирует экземпляр в текущем списке.

Внешние связи

Сервис внешних связей предназначен для улучшения инкапсуляции объекта. Сервис реализует **Принцип клеммника** - все внешние связи объекта устанавливаются через специальный функциональный блок, играющий роль повторителя. Кроме того, при активации сервиса на закладке Внешние связи появляется таблица с полным перечнем внешних связей, и предоставляющих дополнительный функционал – возможность ручного добавления входов, выходов, групп, настройки шкал переменных, а также экспорта и импорта внешних связей. Активация сервиса возможна как до установления связей объекта, так и после того как связи уже были установлены. Если объект уже имеет связи, то активация сервиса приводит к разрыву всех связей, добавлению функционального блока, и автоматическому установлению

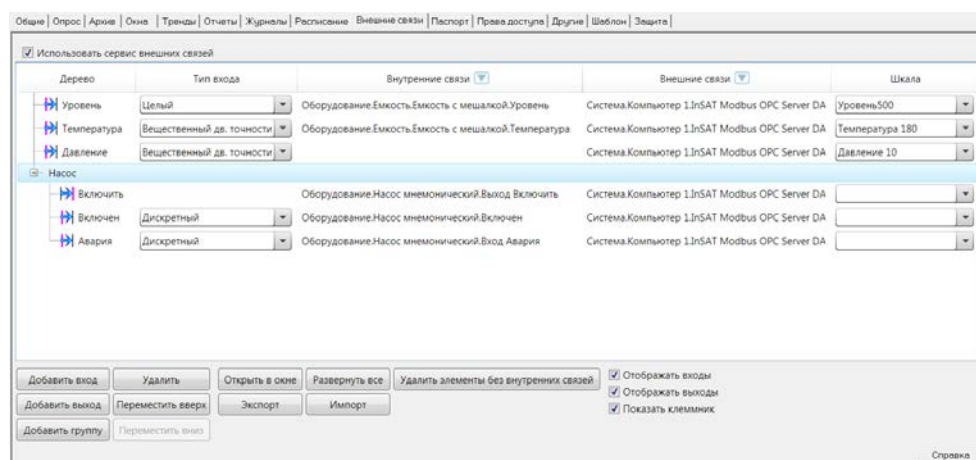
новых связей через входы и выходы ФБ. При наличии внешних связей, создается два функциональный блока – **Входы**, находящийся в верхней части объекта и транслирующий сигналы поступающие в объект, и **Выходы** находящийся в нижней части объекта и передающий сигналы из объекта. Установка новых связей осуществляется стандартными способами – перетаскиваем одной переменной на другую. Связь в клеммнике при этом создается автоматически.

Создаваемый функциональный блок содержит две группы внешние и внутренние, содержащие входы и выходы. Функциональный блок играет роль повторителя – он осуществляет передачу сигнала от входа к соответствующему ему выходу. Группа внешние предназначена для установления связей с внешними элементами (ОПС переменными, выходами других объектов и т.д.). Группа внутренние предназначена для установления связей в внутренними переменными объекта.



Выходы функционального блока являются стандартными выходами MasterSCADA, и имеют возможность задания всех настроек – задания шкалы, настроек архивирования, контроля значений, их также можно использовать в документах – вытаскивать на мнемосхемы, тренды, источники данных отчетов.

Сервис внешних связей (закладка Внешние связи) предназначен для улучшения инкапсуляции объекта - все внешние связи объекта устанавливаются через специальный функциональный блок, играющий роль повторителя. Активизация сервиса возможна до установления связей объекта, так и после того как связи будут установлены. После активации сервиса появляется таблица со списком всех внешних связей объекта. С помощью таблицы можно осуществлять просмотр связей, а также изменять тип входа, пути к связям, добавлять входы, выходы, группы, выполнять функцию экспорта и импорта.

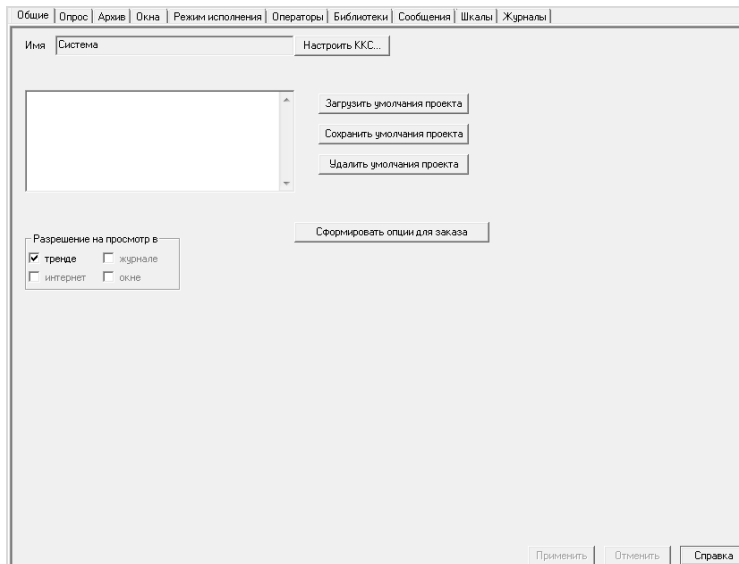


Кнопки и свойства таблицы на предоставленном рисунке соответствуют своим названиям. отдельно можно выделить **Импорт** таблицы внешних связей из формата csv. Данная функция позволяет быстро сформировать таблицу внешних связей используя внешние редакторы (например Excel). Для получения корректной структуры csv файла, рекомендуется вручную установить одну связь в объекте, сделать экспорт таблицы, после чего править сформированный файл в стороннем редакторе.

Страницы свойств Системы

Как правило, свойства системы настраивают для того, чтобы задавать параметры для всего проекта в целом, а так же для задания форм, видов, типов используемых компонентов проекта. Часть настроек системы существует только для того, чтобы автоматически быть унаследованными подчиненными элементами. Таким настройкам имеет смысл устанавливать наиболее часто используемые значения.

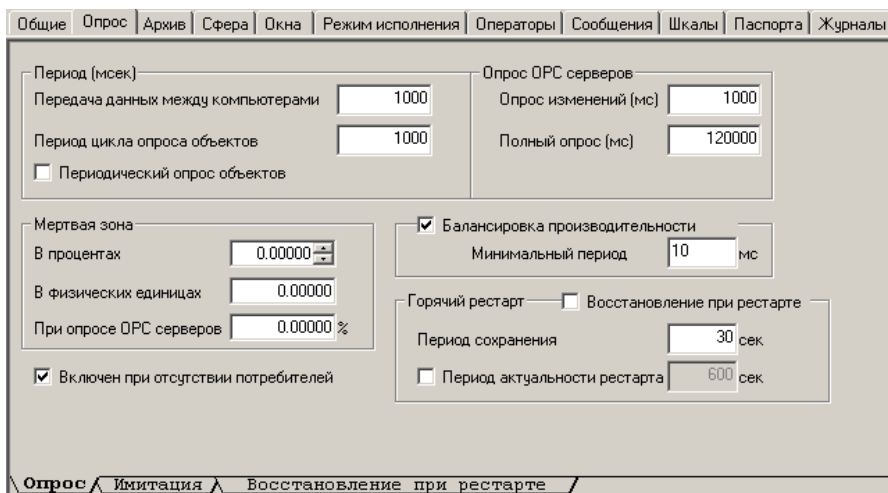
Закладка **Общие** служит для изменения общих свойств Системы.



Элементы управления:

Элемент	Описание
Поле Имя	Имя Системы изменить невозможно.
Поле Комментарий	В случае необходимости можно записать информацию о системе.
Флаг Разрешение просмотра в... (тренде, окне, журнале, интернете)	Позволяет разделить область видимости объекта для различных мест. Например, в тренде данный объект должен быть виден, а в интернет - клиенте, в журнале, в окне - нет.
Кнопка Сформировать опции для заказа	Показывает опции в текущей конфигурации, которые необходимы для работы проекта.
Кнопка Загрузить умолчания проекта	Загружаются настройки проекта по умолчанию (при создании нового проекта используются настройки по умолчанию).
Кнопка Сохранить умолчания проекта	Сохраняются настройки проекта, которые будут загружаться при нажатии на кнопку "Загрузить" умолчания проекта.
Кнопка Удалить умолчания проекта	Удаляются ранее сохраненные умолчания.

На закладке **Опрос** можно установить нужные значение и все элементы, у которых эти значения не были изменены на собственных страницах свойств, унаследуют значения системы.



В группе **Период** задаются настройки опроса переменных:

- Поле передача данных между компьютерами - минимальное время передачи данных между элементами системы, расположенными на разных физических компьютерах;
- Поле период цикла опроса объектов - минимальное время передачи данных между элементами системы, расположенных на одном компьютере;
- флаг периодический опрос объектов - если установлен, то переменным и функциональным блокам назначается периодический способ опроса. Если не установлен - опрос по изменению входов.

В группе опрос OPC серверов настраиваются параметры опроса OPC Серверов:

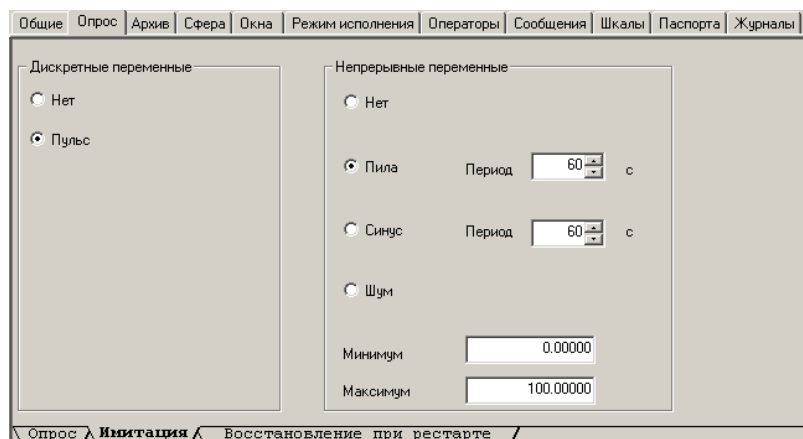
- Поле опрос изменений - минимальный период, с которым Система проверяет OPC переменные на предмет изменения значений;
- Поле полный опрос - период, с которым Система производит полный опрос OPC переменных (в этом случае будет выявлен сбой OPC сервера, если таковой случится). Также будет производиться попытка переподключения к переменным, с которыми по каким-либо причинам обнаружился обрыв связи.

В группе мертвая зона задаются значения мертвой зоны:

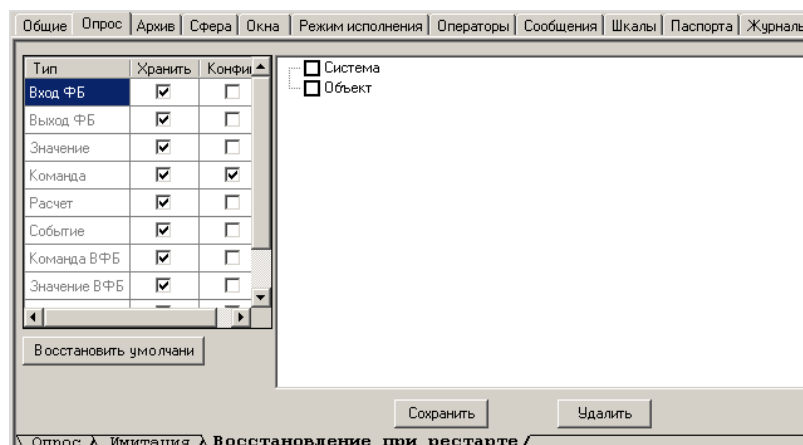
- Поле в процентах - позволяет установить для переменных величину мертвой зоны в процентах от диапазона изменения (который устанавливается на странице свойств переменной общие);
- Поле в физических единицах - величина мертвой зоны для переменных устанавливается в абсолютных величинах;
- Поле при опросе OPC серверов" - в поле указывается величина мертвой зоны, используемая при запросе данных OPC сервера по подписке.

Назначения флагов балансировка производительности, включён при отсутствии потребителей и группа горячий рестарт будут более подробно описаны в последующих главах.

На листе **Имитация** устанавливаются параметры имитации законов изменения значений входов. Это будет касаться только тех входов, у которых нет связей и установлена кнопка имитация.



На листе Восстановление при рестарте настраиваются восстановления для разных типов параметров.



В столбце **хранить** указывается необходимо ли хранить элементы данного типа, если задана настройка **Восстановление при рестарте** у родительского элемента. У конкретного элемента можно будет переопределить данную настройку. В столбце **конфигурация** указывается, что значения для восстановления нужно хранить в специальном файле. При работе в контроллере эти значения будут включены в конфигурацию контроллера или в отдельный файл `values.bin`. Если напротив команды снять флаг **конфигурация**, то для всех команд, вложенных в объект, значения будут храниться в файле рестарте и на них будет действо-

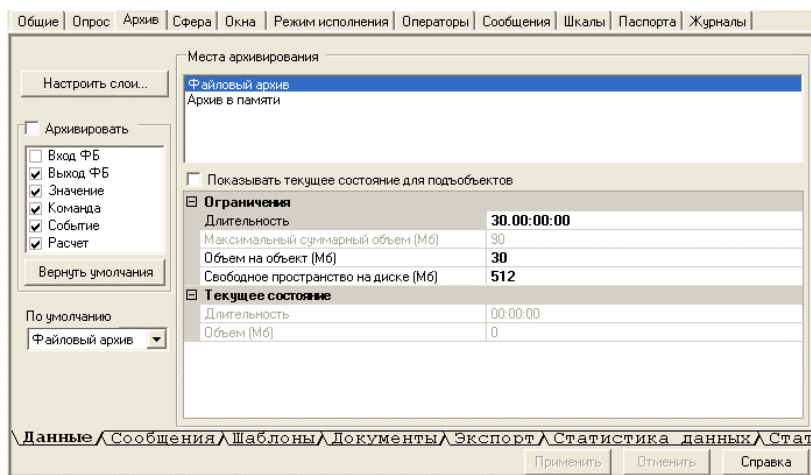
вать настройка *Актуальность рестарта*. Если флаг установлен, то настройка актуальность рестарта не будет действовать.

На закладке **Архив** определяются базовые настройки архива данных, такие как место хранения, время хранения и т.п. MasterSCADA позволяет хранить архивы в собственной файловой системе, в оперативной памяти компьютера, либо во внешней базе данных.

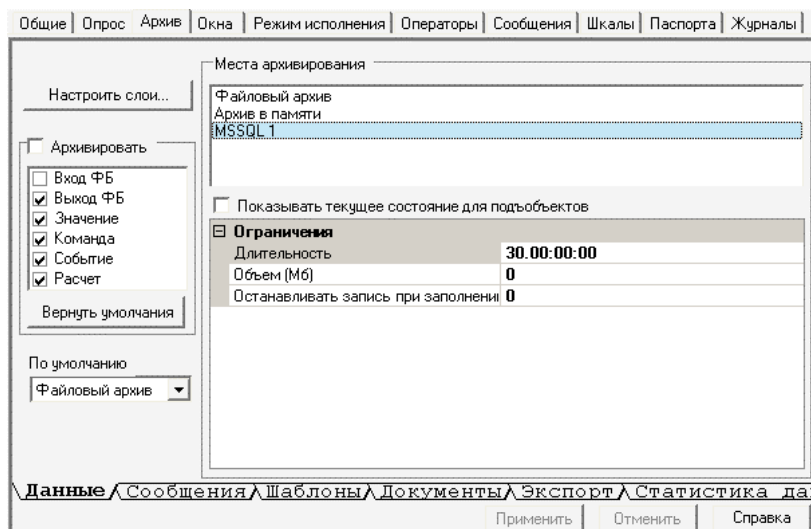
Файловая система - архивы данных хранятся локально в двоичных файлах в собственном формате, доступ к этим архивам можно получить через стандартные функции и документы MasterSCADA (тренды, журналы, функциональные блоки для работы с архивом, функции расчета). Для использования этой возможности архивирования дополнительные опции не требуются.

Оперативная память - служит для временного хранения данных. Используется в случае, если необходимо архивировать параметр скорость изменения которого выше, чем скорость записи в стационарные архивы (в файловый архив и во внешнюю БД). Также оперативная память используется, если параметр был помещен на тренд в режиме исполнения и в настройках проекта архивирование этого параметра не было предусмотрено.

База данных - архивы данных помимо внутренних архивов могут храниться во внешних базах данных. В дальнейшем список используемых баз данных для хранения архивов будет расширен. В рамках одного проекта можно использовать несколько баз данных. Например, технологические параметры могут попадать в одну базу данных, а рассчитанные экономические - в другую.



При использовании внешних баз данных окно имеет следующий вид.



Для оптимизации работы с данными за большие интервалы времени применяются слои. Изначально формируется основной слой - результат работы MasterSCADA, куда попадают все данные. Остальные слои образуются в результате обработки основного. Данные хранимые в архиве могут прореживаться в зависимости от слоя: прореженные данные за каждую минуту, за каждый час, сутки, либо за произвольный интервал времени.

Поле архивировать определяет умолчания для дочерних элементов, значения каких элементов проекта будут архивироваться. В дальнейшем для каждого отдельного элемента деревьев настройку можно переопределить

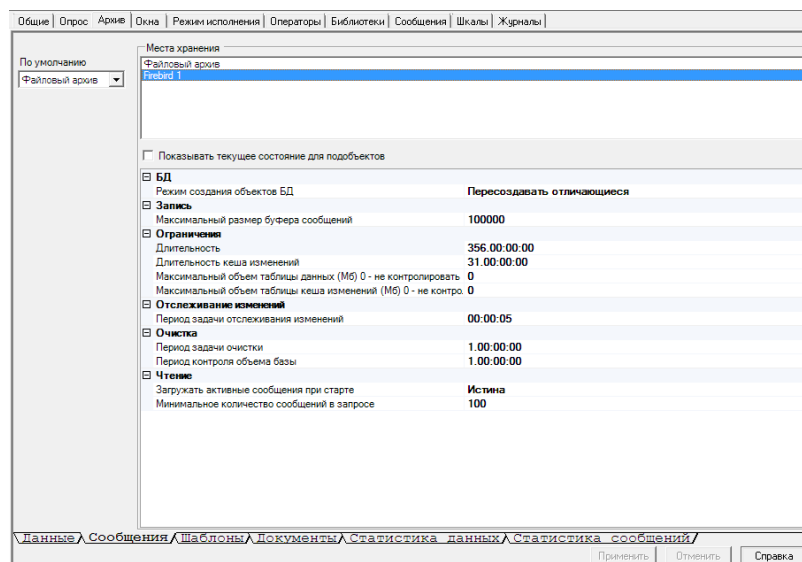
Ограничения:

Место хранения	Название	Описание
Файловый архив	Длительность	Ограничивает архив по времени. Формат записи: дни.чч:мм:сс.
	Максимальный суммарный объем	Ограничивает размер суммарного архива. Настраивается только в том случае, если место хранения данных внешняя БД.
	Объем на объект	Ограничивает размер архива для каждого из объектов. Настраивается в случае, если место хранения архивов - файловая система.
	Свободное пространство на диске	Если на диске остается места меньше, чем указано, то архив MasterSCADA перестает расти. По мере поступления данных самые ранние будут удаляться. Настраивается в случае, если место хранения архивов - файловая система.
Внешняя база данных	Объем (Мб)	Ограничивает размер архива при архивировании в базу данных.
	Останавливать запись при заполнении более чем (%)	При заполнении базы данных более указанного объема, запись останавливается. Настраивается в случае, если место хранения архивов - внешняя база данных.

Лист **Сообщения** определяет базовые настройки архива сообщений, такие как место хранения, время хранения и т.п. MasterSCADA позволяет хранить архивы сообщений в собственной файловой системе, в оперативной памяти компьютера, либо во внешней базе данных.

Таблица данных - основное хранилище сообщений, а также дополнительной информации по ним (время квитирования, время, когда событие потеряло актуальность и т.п.).

Таблица кеша изменений - это таблица, в которую пишутся все изменения в таблице данных (новое сообщение, времена квитирования, закрытия и т.п.) это служебная таблица, служит для оптимизации работы MasterSCADA с сообщениями. Она нужна для решения получения всех активных сообщений за определённый период, например, если включается компьютер-клиент в систему, с периодом в 1 секунду и для просмотра изменений таблицы данных.



Настройки базы данных сообщений:

Название	Описание
Максимальный размер буфера сообщений	Максимальное количество изменений, которые могут накопиться в буфере, до того момента как изменения будут записаны в базу.
Длительность	Ограничивает архив по времени. Формат записи: дни.чч:мм:сс. При наступлении ограничений более старые данные будут удаляться.
Длительность кеша изменений	Это ограничение на таблицу БД, которая используется для передачи сообщений между компьютерами.

Название	Описание
Максимальный объем таблицы данных (Мб) 0 - не контролировать	При достижении указанного объема более старые данные будут удаляться из таблицы данных.
Максимальный объем таблицы кеша изменений (Мб) 0 -не контролировать	При достижении указанного объема более старые данные будут удаляться из таблицы кеша изменений.
Период задачи отслеживания изменений	Интервал времени, с которым запускается проверка на изменения.
Период задачи очистки	Интервал, с которым запускается проверка на переполнение таблицы по времени. Если время хранения базы будет преувеличено, то произойдет очистка.
Период контроля объема базы	Интервал, через который происходит проверка на переполнение таблицы по объему. Если размер таблицы больше указанного, то произойдет очистка.
Минимальное количество сообщений в запросе	Количество сообщений передаваемое из БД в MasterSCADA для отображения в журналах и т.п.

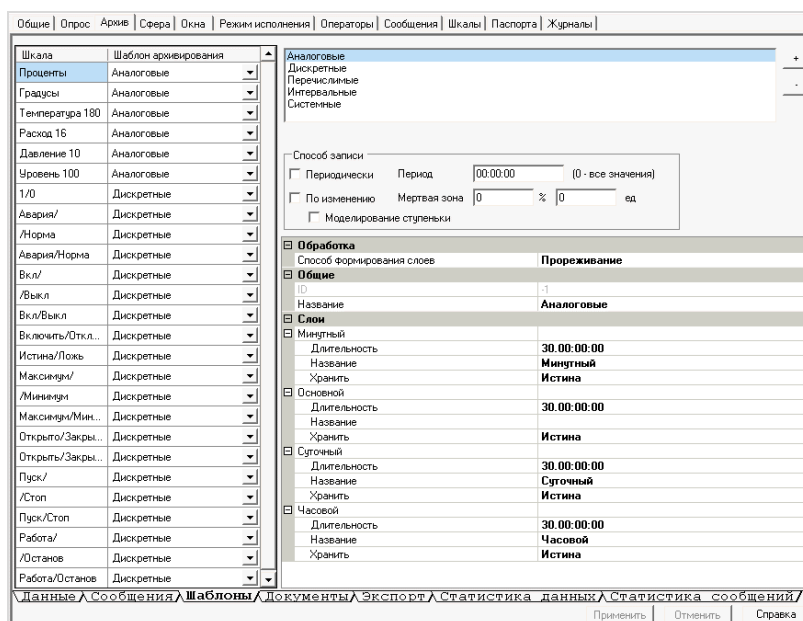
Лист **Документы** определяет наиболее часто встречающиеся размеры архивов для наследования. Страница свойств настраивается в случае использования файлового архива MasterSCADA.

Название	Длительность (дней)		Объем (Мб)		
	Задаю	Есть	На объект	Есть	Всего
Сохранение мнемосхем	30		3.00		
Сохранение тренды	30		3.00		
Сохранение рапорты	30		3.00		
Сохранение журналы	30		3.00		

Подсчет суммарного размера

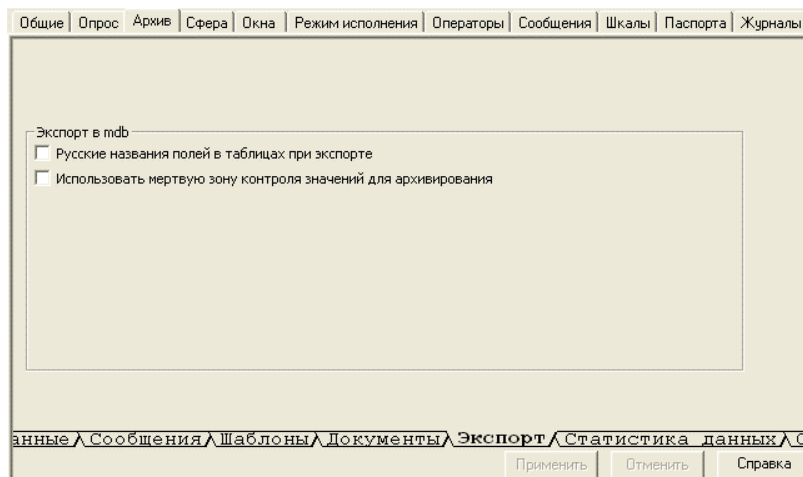
В элемент управления таблица следует вписать длительность хранения и максимальный объем каждого элемента архива. Поля *Есть* заполняются при нажатии кнопки *Подсчет суммарного размера*. Если сработало одно из ограничений, то самые первые данные, поступившие в архив, будут удаляться. Строки сохраненные мнемосхемы, сохраненные тренды, сохраненные рапорты, сохраненные журналы - настраиваются только в том случае, если необходимо сохранять и просматривать документами сторонними программами. Элемент подсчет суммарного размера подсчитывает суммарный размер архивов всех объектов и компьютеров.

Лист **Шаблоны** служит для создания шаблонов архивирования. Шаблон архивирования назначается каждому интервалу измерений, введенному на странице свойств системы шкалы.

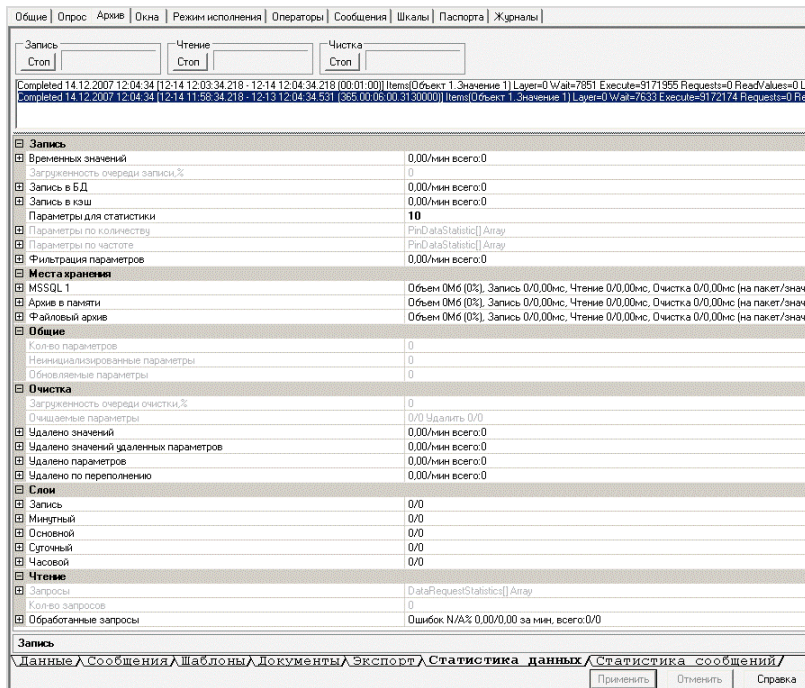


Страница свойств визуально разделена на две части. В правой части создаются и настраиваются шаблоны. В левой части назначается шкале определенный шаблон, который будет использоваться для переменных данного типа по умолчанию. В дальнейшем для каждой конкретной переменной шаблон архивирования может быть переопределен. Для создания нового шаблона архивирования необходимо нажать кнопку с обозначением +. Затем настроить свойства в таблице. Свойства шаблонов более подробно описаны в справке MasterSCADA. Для удаления существующего шаблона нужно нажать кнопку с обозначением -. После создания шаблона его требуется присвоить нужному диапазону измерения в левой части страницы свойств, используя выпадающий список.

Лист **Экспорт** настраивается в случае использования экспорта в БД MasterSCADA.



Лист **Статистика данных** служит для отображения статистических данных работы архива по данным в режиме исполнения.



На закладке **Окна** устанавливается высота и ширина окон документов, добавляются и удаляются новые типы окон. Эти настройки наследуются компьютерами, для которых впоследствии можно будет переопределить значения всех атрибутов окон (кроме значка). У компьютеров эта закладка почти такая же. Создание же конкретных документов настроенных здесь типов производится для объектов на одноименной закладке свойств объекта, для которого нужно создать документ.

Документы, предопределенные в системе:

- **Мнемосхема** - это динамизированная технологическая схема объекта, которая может открываться в Windows-окне или на полном экране;
- **Окно объекта** - динамизированная технологическая схема объекта, открываемая в всплывающем окне поверх мнемосхемы другого объекта. Как правило, включает в себя в качестве составной части данные объекта;
- **Окно управления** - всплывающее окно, содержащее органы управления объектом и открывающееся поверх мнемосхемы другого объекта;
- **Рапорт** - документ, предназначенный для печати или просмотра и содержащий текущие значения технологических параметров на определенный момент времени, или результаты (средние, интегральные и т. п.) значений технологических параметров за определенный период времени;
- **Журнал** - средство для просмотра сообщений, в котором реализована возможность фильтровать сообщения по тем или иным признакам;
- **Тренд** - предназначен для просмотра данных в графическом и табличном виде;
- **Схема** - документ предназначен для редактирования схемы функциональных блоков;
- **Изображение объекта** - документ предназначен для создания пользовательских изображений объекта, создается как обычная мнемосхема. Может быть использовано, как окно, либо как символ объекта. При вставке изображения объекта на мнемосхемы других объектов сохраняются все его свойства и настройки динамизации.

	Название	Ширина	Высота	Тип	Свойства	Все
	Мнемосхема	1024	736	По выбору	...	<input type="checkbox"/>
	Окно объекта	200	150	Окно	...	<input type="checkbox"/>
	Окно управления	200	150	Окно	...	<input type="checkbox"/>
	Рапорт	1024	736	По выбору	...	<input type="checkbox"/>
	Рецепт	1024	736	По выбору	...	<input type="checkbox"/>
	Изображение объекта	200	150	Окно	...	<input type="checkbox"/>
	Журнал	800	600	По выбору	...	<input type="checkbox"/>
	Тренд	800	600	По выбору	...	<input type="checkbox"/>
	Схема	800	600	По выбору	...	<input type="checkbox"/>

Добавить | Удалить | Установить стандартные иконки

На закладке **Операторы** настраивается разграничение прав доступа для тех или иных групп пользователей. Лист **Права доступа** выглядит следующим образом.

Должность		Старший оператор	Лок.	Глоб.	Журнал	Подтве
Категория	Действие					
Проект	Включить режим исполнения	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Отключить режим исполнения	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Навигация в дереве системы	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Навигация по объектам	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Перейти к проекту	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Минимизировать окно проект	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Экспортировать архив	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Открыть журнал действий	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Изменение списка операторо	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Доступ через Internet	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Окно сообщений	Окно управления системой	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Журнал неактивных соо	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Открыть окно	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Права на действия с документами можно настраивать как на все документы, устанавливая и снимая флажки напротив названия действия, так и переопределять для конкретного типа документа. Установкой и снятием флажков в соответствующих местах настраиваются права доступа.

- Лок. - доступно на одном компьютере;
- Глоб. - доступно на любом компьютере;
- Журнал - записывать о совершенном действии в журнал;
- Подтверждение - перед выполнением действия запрашивать подтверждение.

На листе **Смены** возможно добавить и удалить смену, а также изменять название и время смены. Времена смен не должны перекрываться. Вход оператора возможен не ранее, чем за полчаса до начала его смены. Выход оператора возможен в любое время по кнопке **Завершение сеанса**.

На листе **Должности** возможно настраивать список должностей операторов. Назначение реальных операторов для каждого компьютера производится в свойствах компьютера (закладка операторы). Флаг **Посменно** указывает на то, что оператор с данной должностью должен быть назначен работать в определенные смены, если же флаг не установлен, то оператор может входить в систему в любое время. При добавлении новой должности она получает права доступа той должности, которая была выделена в момент вставки.

На закладке **Сообщения** настраивается предоставление информация для оператора в текстовом виде. На листе **Категории** производятся настройки числа категорий, их каналов вывода и цветов.

Категория	Приоритет	Канал
Авария	Максимальный	<input checked="" type="checkbox"/> Сообщение
Предупреждение	Высокий	<input checked="" type="checkbox"/> Строка статуса
Отсутствие значения	Средний	<input checked="" type="checkbox"/> Журнал
Информация	Средний	<input type="checkbox"/> Принтер
Временная неисправность	Средний	<input checked="" type="checkbox"/> Архив
Взрыв	Максимальный	
Армагеддон	Обычный	

Журнал	
<input checked="" type="checkbox"/>	Основной журнал
<input checked="" type="checkbox"/>	Аварийные сообщения
<input type="checkbox"/>	Инфо для операторов
<input type="checkbox"/>	Системные сообщения

Звук...

Добавить Удалить Пример текста сообщения Текст Фон

Категории Авария, Предупреждение и Отсутствие значения существуют по умолчанию и не удаляются. В столбце категории можно добавлять, удалять и переименовывать добавленные категории сообщений. В столбике приоритет из всплывающего списка выбирается один из четырех приоритетов: Максимальный, Очень Высокий, Высокий, Средний, Обычный, Низкий.

Список каналов содержит перечень каналов вывода сообщений, которые можно включать и отключать установкой и снятием соответствующих флажков слева от названия канала. Пе-

речень соответствует той категории, которая в данный момент выделена в таблице и наследуется сообщениями этой категории. Для каждой категории в полях цвет текста и цвет фона можно выбрать соответствующие цвета. Как и список каналов, текущие цвета относятся только к той категории, которая в данный момент выделена в таблице. Нажатие кнопки звук позволяет выбрать .wav файл. Выбранным звуком будет сопровождаться появление сообщения в окне сообщений.

Лист **Каналы** служит для настройки каждого канала в отдельности. Для выбора канала для настройки существует всплывающий список в левом верхнем углу. Сняв флажок использование канала разрешено имеется возможность запретить канал.

Канал	Описание
Окно сообщений	Окно сообщений появляется в режиме исполнения и содержит информацию для оператора. При настройке всплывтия окна слово активный объект означает - объект, выделенный в дереве в данный момент.
Строка статуса	Строка статуса - это текст внизу окна программы.
Журнал	Описание настроек журнала.
Принтер	Сообщение, направленное в этот канал будет распечатываться на принтере.
Архив	Сообщения, направленные в архив сохраняются в файле после окончания.
SMS	Сообщение, направленное в этот канал будет передаваться через GSM-модем в виде SMS-сообщения на номер указанного абонента.
E-mail	Сообщение, направленное в этот канал будет отослано по электронной почте на указанный адрес, при наличии связи с Интернетом.

Лист **Сообщения** выглядит следующим образом.

Категория	Сообщение
<input checked="" type="checkbox"/> Предупреждение	На диске осталось мало места
<input checked="" type="checkbox"/> Информация	Вход в систему
<input checked="" type="checkbox"/> Информация	Смена даты
<input checked="" type="checkbox"/> Предупреждение	Связь с компьютером не установлена
<input checked="" type="checkbox"/> Отсутствие значения	Обрыв связи с компьютером
<input checked="" type="checkbox"/> Информация	Отключение компьютера
<input checked="" type="checkbox"/> Информация	Формирование рапорта
<input checked="" type="checkbox"/> Предупреждение	Ошибка принтера
<input checked="" type="checkbox"/> Отсутствие значения	Нарушена связь с OPC сервером
<input checked="" type="checkbox"/> Отсутствие значения	Ошибка в OPC сервере
<input checked="" type="checkbox"/> Предупреждение	Конфигурация OPC сервера не загружена
<input checked="" type="checkbox"/> Предупреждение	OPC сервер приостановлен
<input checked="" type="checkbox"/> Информация	OPC сервер функционирует в тестовом режиме
<input checked="" type="checkbox"/> Предупреждение	Система не успевает обработать данные от OPC сервера

Системные
 Функциональных блоков
 Переменных

Настройки окна сообщений:

Название	Назначение
Всплывать в любом случае	
Всплывать, если активен родной объект	
Всплывать если активен родной или дочерний объект	Определяет условие появления окна сообщений в режиме исполнения.
Всплывать если активен родной или родительский объект	
Не всплывать	
Область Положение	Определяет координаты левого верхнего угла сообщения, ширину его и высоту. Параметры: Левый -1, Верхний -1, Ширина 0, Высота 0, означают размер и местоположения окна по умолчанию.
Число сообщений в буфере	Количество сообщений, которые можно будет просмотреть в окне сообщений. Если сообщений выводилось больше указанного значения, то в окне сообщений будут отображаться последние из них. Эта настройка не относится к системе архивирования, влияет только на отображение сообщений в данном окне.
Флаг Использовать цвет	Если флаг установлен, то сообщения, для категорий которых назначен цвет на странице свойств системы сообщения/ категории будут отображаться в цвете.
Флаг Использовать звук	Если флаг установлен, то сообщения, для категорий которых назначен звук на странице свойств системы сообщения/ категории будут сопровождаться звуком. Если компьютер не снабжен динамиками, наушником и т.п. настройка данная не имеет значения.
Флаг Повтор звука	Флаг имеет смысл устанавливать если установлен флаг использовать звук. Если флаг повтор звука не установлен, то мелодия проиграется

Название	Назначение
Отображаемые поля	<p>один раз. Если флаг установлен, то мелодия будет проигрываться до тех пор, пока оператор либо не квитирует сообщение, либо не закроет окно сообщений, либо пока не пропадут условия возникновения сообщения. Настройка не действует для тех сообщений, которые не требуют квитирования.</p> <p>Источник - в окно отображается источник сообщения; Комментарий - в окно отображается комментарий переменной; Без экземпляра - если настройка отключена, то в поле источника также указывается имя экземпляра.</p>
Выбрать шрифт	Открывает диалоговое окно, в котором настраивается тип шрифта. Параметры шрифта курсив, жирный не настраиваются. Эти настройки показывают, какие из сообщений не были квитированы, а какие еще не потеряли свою актуальность.

Настройка канала *E-mail*:

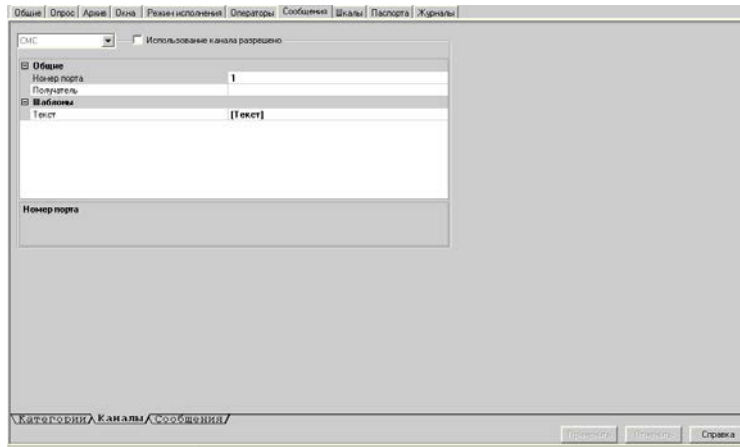
Свойство	Назначение
Имя учетной записи	Логин доступа на SMTP сервер.
Пароль	Пароль доступа на SMTP сервер.
Отправитель	Электронный адрес отправителя.
Получатель	Электронный адрес получателя.
Сервер SMTP	Используемый SMTP сервер.
Текст	Задается шаблон текста письма.
Тема	Задается шаблон темы письма.

Текст письма может содержать любое свойство сообщения: текст, категорию, время, источник и т.п. Для того, чтобы свойство сообщения попало в текст письма, отправляемого MasterSCADA необходимо указать это свойство в квадратных скобках: []. Если необходимо вставить какой-либо произвольный текст в письмо, то текст набирается в шаблоне без скобок. Если необходимо написать что-либо с новой строки, то необходимо поставить в шаблоне \. Возможные свойства сообщения и их способы отображения описаны в справке MasterSCADA. Сообщения отправляются при наличии устойчивой сети. В случае невозможности отправки E-mail, сообщения не накапливаются.

Настройки канала *Принтер*:

Функции выполняемые при установке флагов соответствуют наименованиям флагов. Сообщения распечатываются, когда их объём составляет целый лист, но по истечении этого времени (после печати последнего листа) сообщения будут напечатаны даже в том случае, если на лист их еще не набралось.

Настройка канала SMS:



Настройка	Назначение
Номер порта	Указывается com-порт компьютера, к которому подключен GSM-модем.
Получатель	Указывается номер телефона получателя сообщений. Номер телефона задается только в режиме разработки.
Текст	Задается шаблон сообщения, аналогично как и для канала вывода сообщений E-mail. При составлении шаблона желательно, чтобы получившееся исходящее сообщение уместилось в одну SMS-посылку. В противном случае, сотовый оператор разделит ваше сообщение на несколько, что может ввести в заблуждение получателя.

Закладка **Шкалы** позволяет настроить необходимые шкалы для их последующего применения для различных элементов объекта. Лист **Аналоговые** выглядит следующим образом

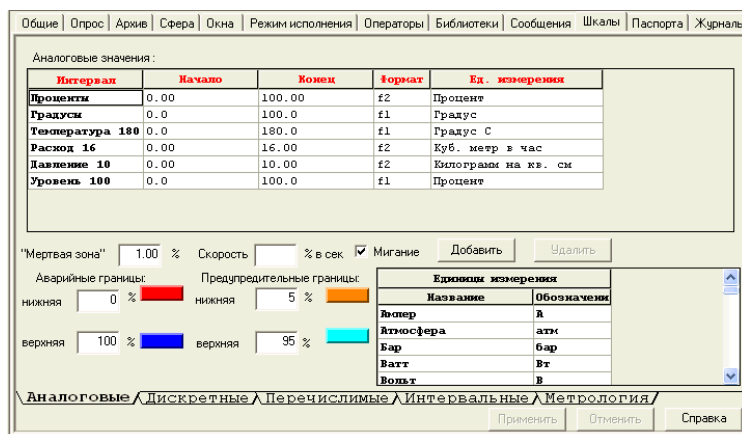


Таблица Аналоговые значения:

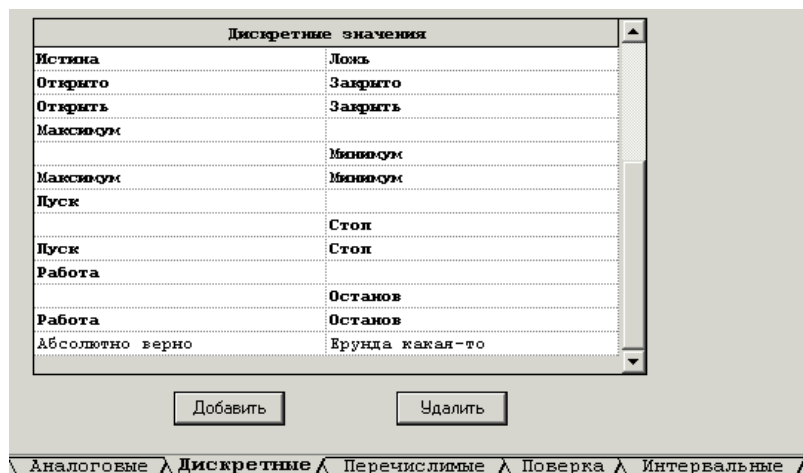
- **Интервал** - название интервала;
- **Начало** - нижняя граница интервала (от нее вычисляются нижняя аварийная и предупредительная границы);
- **Конец** - верхняя граница интервала (от нее вычисляются верхняя аварийная и предупредительная границы);
- **Формат** - в этом поле задается формат отображения значения. Буква обозначает тип отображения: **f (float)** - число с плавающей запятой (например 12,7234). **e (exp)** - экспоненциальный формат (например $1,2 \cdot e^{-10}$). Число обозначает количество отображаемых знаков после запятой;
- **Единица измерения** - выбирается из всплывающего списка.

В программу встроено несколько десятков единиц измерения. В случае необходимости имеется возможность добавления пользовательских единиц измерения. Поля группы **Границы** относятся к текущему, (т.е. выделенному в настоящий момент в рамочку) интервалу в таблице **Аналоговые значения**. Существуют следующие поля:

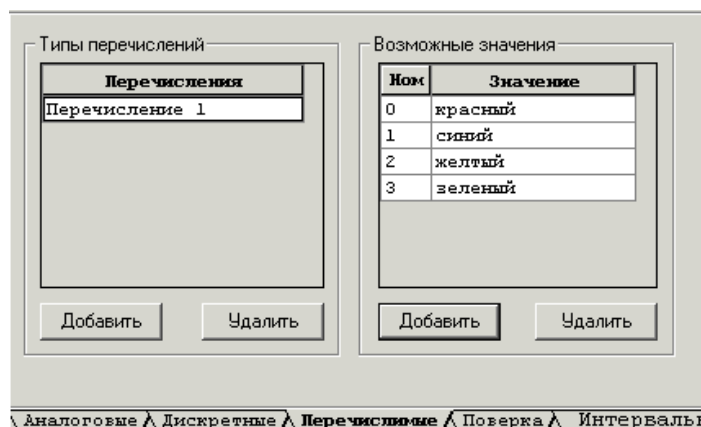
- Мертвая зона;
- Скорость изменения;
- Верхняя и нижняя аварийные границы;
- Верхняя и нижняя предупредительные границы.

Значения задаются в процентах. На основе данных таблицы и этих параметров будут автоматически вычислены абсолютные значения соответствующих параметров. Данную настройку нужно использовать в случае, когда несколько элементов (например, выходов объектов) имеют одинаковые значения границ. Тогда достаточно будет указать соответствующий интервал, и все значения выставятся автоматически. Цветные кнопки служат для настройки цвета отображаемого в проекте значения переменных (входов и выходов) в пределах соответствующих границ.

Лист **Дискретные** содержит пары дискретных значений. Имеется возможность добавить пользовательские названия.



Лист **Перечислимые** используется для настройки шкал данного типа.

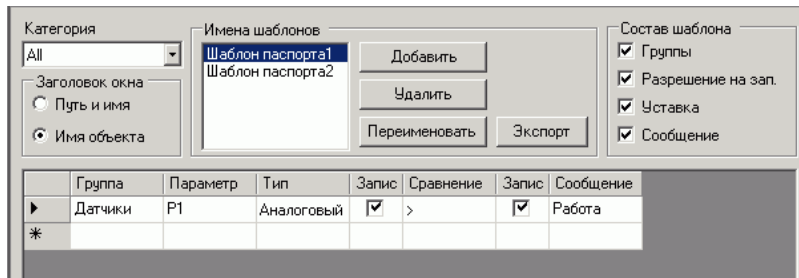


Список возможные значения содержит значения перечисления. В этой таблице показываются значения, относящиеся к выделенному в данный момент перечислению в таблице Типы перечислений. С помощью кнопок **Добавить** и **Удалить**, находящихся внизу таблицы, редактируется состав текущего перечисления. Каждая запись в таблице Возможные значения состоит из двух полей - числового значения, и соответствующей этому значению строки. В режиме исполнения везде, где отображается значение переменной (дерево объекта, мнемосхема, и т.п.) будет показываться не числовое, а соответствующее строковое значение. **Интервальные** шкалы настраиваются аналогично.

Закладка **Паспорт**. Основное назначение Паспортов - предоставление оператору справочной информации об объектах и оборудовании. Паспорт - это таблица параметров, описывающих объект. Значение некоторых параметров задается на этапе разработки, другие получают значения в режиме исполнения от переменной или в результате их ввода оператором. Значения параметров могут сравниваться с уставками, в результате чего в режиме исполнения могут формироваться сообщения. Подсистема паспортизации в MasterSCADA основывается на следующих понятиях:

- **Шаблон паспорта** - определяет набор параметров паспорта (количество, имена и типы), а также позволяет задать значения всех полей, за исключением Значения и Уставки, которые будут установлены по умолчанию для всех экземпляров паспорта данного шаблона;
- **Экземпляр паспорта** - создается на основе шаблона паспорта путем ввода значений параметров и их уставок. Все остальные поля предопределены шаблоном паспорта.

Паспорта могут иметь Объекты, Визуальные функциональные блоки, а также все типы переменных MasterSCADA. Шаблоны паспортов формируются на странице свойств системы паспорта.



Список выбора **Категория** - определяет элементы проекта, для которых может быть использованы создаваемые шаблоны. Группа **Имена шаблонов** содержит элементы управления, необходимые для создания, удаления и переименования шаблонов паспортов в выбранной категории. Список имена шаблонов отображает имена шаблонов в выбранной категории. Группа **Состав шаблона** - определяет состав полей параметров шаблона. **Таблица параметров шаблона** - каждый из параметров, описываемых в шаблоне имеет набор полей, приведенных в таблице:

Поле	Описание	Переопределяется или нет в экземпляре
Группа	Служат для группировки логически связанных параметров.	Нет
Параметр	Имя параметра.	Нет
Тип	Тип параметра. Поддерживаются все типы переменных, имеющихся в MasterSCADA.	Нет
Запись значения	Если флаг установлен, значение параметра можно редактировать в режиме исполнения.	Да
Сравнение	Оператор сравнения значения параметра с уставкой.	Да
Запись уставки	Если флаг установлен, значение уставки можно редактировать в режиме исполнения.	Да
Сообщение	В этом поле можно ввести текст сообщения, которое будет выводиться в режиме исполнения при выполнении условия сравнения значения параметра с уставкой.	Да

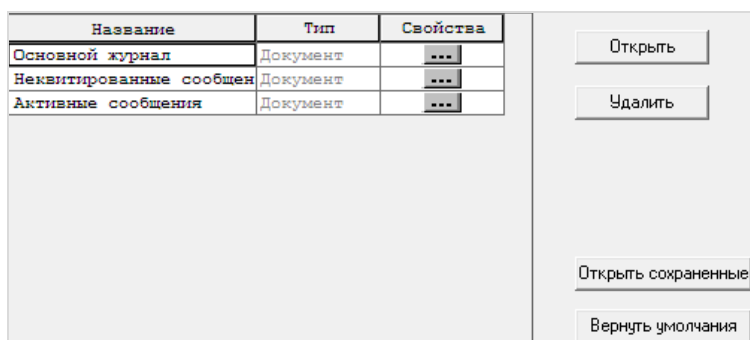
Значения переопределяемых полей (см. таблицу) могут быть уточнены при формировании экземпляра Паспорта. Кнопка **Экспорт** позволяет экспортировать выделенный в списке шаблон в файл, имеющий формат csv, для редактирования которого можно воспользоваться MS Excel. При этом, чтобы файл открылся корректно, необходимо в диалоговом окне открытия файлов в Excel указать тип файлов - *Текстовые файлы*. В проэкспортированном файле можно изменять значения только переопределяемых полей. Впоследствии отредактированный csv-файл с определением шаблона можно использовать для создания экземпляров паспортов.

Флаг **Включен при отсутствии потребителей**. Отсутствием потребителей у элемента считается ситуация, в которой:

- Элемент не имеет ни одной переменной, связанной с окном (мнемосхемой, и т. д), которое в настоящий момент является открытым. Связи с закрытыми окнами не считаются;
- Элемент не имеет ни одной переменной, связанной с другим элементом, или все элементы, с которыми связаны переменные, сами являются отключенными при отсутствии потребителей.

Очевидно, что в такой ситуации формирование значений переменных элемента не имеет смысла, так как они нигде не используются. Если есть необходимость экономить ресурсы компьютера, снимите флаг Включен при отсутствии потребителей, тогда элемент не будет формировать никому не нужные данные и занимать этим процессор.

Закладка **Журналы** представлена на рисунке ниже. Таблица содержит перечень доступных журналов. Для каждого журнала можно настроить способ открытия, а также расположение на экране.



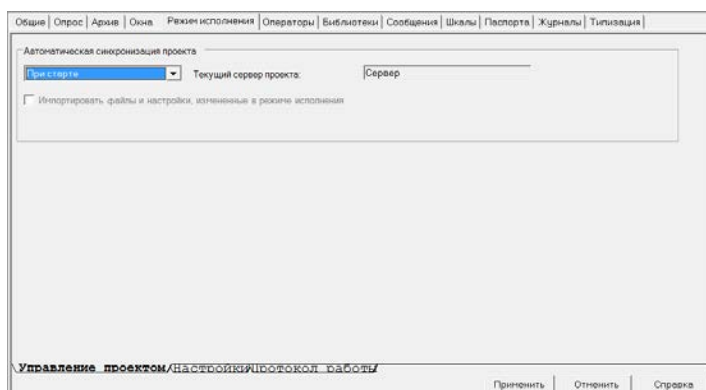
Закладка **Метрологическая поверка** решает следующие проблемы:

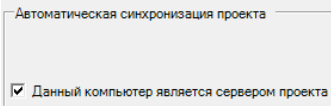
- Упрощает процесс поверки переменных (записи значений);
- Позволяет автоматически создавать и печатать документы для метрологической аттестации.

Работа осуществляется в три этапа:

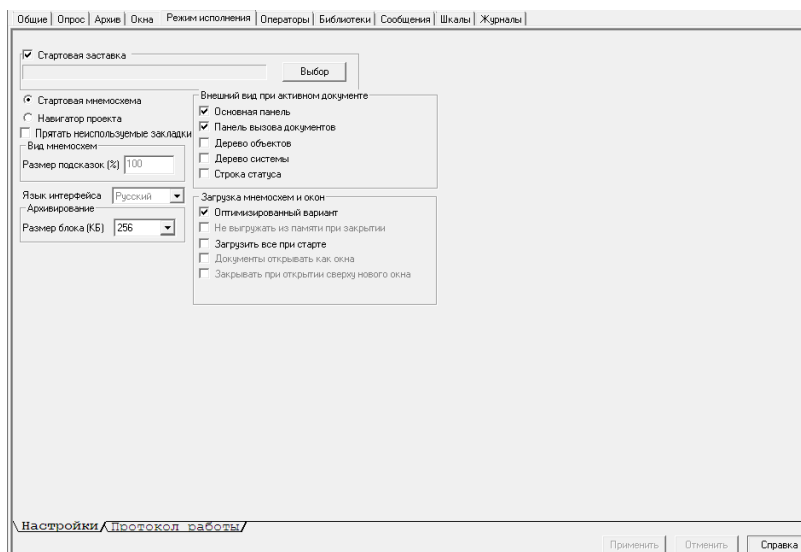
- **Настройка методик поверки.** Каждая методика поверки соответствует группе измерительных каналов и связана со шкалой MasterSCADA. Связь методики и шкалы предоставляет следующие преимущества - часть настроек шкалы (например, минимум и максимум) автоматически становятся настройками группы ИК, чем снимается необходимость в их повторном установлении, а также переменные, которым назначена шкала, автоматически становятся принадлежащими определенной группе ИК. Таким образом, без дополнительных действий, системе известно о количестве ИК в каждой группе. В настройку методики входит задание контрольных точек и задание свойств методики, необходимых для проведения статистических вычислений, или для заполнения определенных полей в аттестационных документах;
- **Поверка переменных.** При поверке записываются значения переменных, соответствующие контрольным точкам методики. Записанные величины помещаются в отдельный архив, из которого считываются впоследствии для выполнения статистических расчетов и помещения в аттестационные документы. Очевидно, что автоматизировать полностью этот процесс не представляется возможным, но с опцией метрологическая поверка делать все становится намного удобнее, чем вручную. У каждой поверяемой переменной есть специальная закладка, на которой отображается список контрольных точек, таблица значений и другая полезная функциональность. Текущие значения переменной записываются одним нажатием кнопки;
- **Формирование отчетов.** Для метрологической аттестации необходимо предоставить определенный пакет документов. Некоторые из них индивидуальны для каждой системы управления и могут быть составлены только с участием разработчиков системы на месте проведения аттестации. Но большинство документов, имеет сходную для всех систем структуру, такие отчеты автоматически формируются с помощью MasterSCADA. В них помещаются сведения о системе, поверяемых переменных, результаты статистических вычислений. MasterSCADA формирует отчеты на основе заранее созданных шаблонов, хранящихся на компьютере в известном месте и имеющих формат, доступный для редактирования. Возможна корректировка шаблонов без изменения содержимого ссылок полей, предназначенных для автоматического заполнения.

На закладке **Режим исполнения** осуществляется настройка автоматической синхронизации сетевого проекта



Поле	Элемент	Назначение
Автоматическая синхронизация проекта	Режим Нет	Автоматически не синхронизируется.
	Режим При старте	Означает, что при запуске проекта будет происходить автоматическая синхронизация проекта.
	Режим При изменении	Означает, что при любом изменении проекта в RT будет происходить синхронизация.
	Импортировать файлы и настройки, измененные в режиме исполнения Данный компьютер является сервером проекта	При синхронизации проекта все изменения сделанные в RT (настройки ,файлы) будут импортированы на клиент. При выставлении галочки компьютер становится сервером.
		
Экспортировать проект	Текстовое поле	Указывает место, куда будет проэкспортирован проект.
	Кнопка Выбрать	Открывает диалоговое окно Windows, которое позволяет выбрать директорию, в которую экспортируется проект.
	Кнопка Экспортировать	Запускает экспорт проекта
Отчет об ошибках	Тестовое поле	
	Кнопка Выбрать	
	Кнопка Сформировать	Формирует отчет об ошибках.
	Включая конфигурацию текущего проекта	Определяет состав отчета об ошибках. Если галочка установлена, то отчет будет содержать текущий проект (накопленные архивы в отчет не попадут)
	Заархивировать	Образовавшаяся папка автоматически заархивируется
	Включая снимок экрана	Определяет состав отчета об ошибках. Если галочка установлена, то отчет будет содержать скриншот экрана монитора на момент формирования отчета.
	Отправить в службу технической поддержки	Если галочка установлена, то отправка отчета об ошибках по электронной почте произойдет автоматически
Импортировать проект	Текстовое поле	Указывается место нахождения импортируемого проекта
	Кнопка Выбрать	Открывает диалоговое окно Windows, которое позволяет выбрать директорию, в которой хранится импортируемый проект.
	Кнопка Импортировать	Запускает импорт проекта
	Импортировать настройки, измененные в режиме исполнения	
	Не показывать диалоги в режиме исполнения	
Текущий сервер проекта	Сервер	Компьютер, выбранный в качестве сервера проекта (компьютер с "эталонным" проектом).

Лист настройки выглядит следующим образом.



Элементы управления на данном листе отвечают за следующие действия:

Элемент	Описание
Стартовая заставка	Предоставляется возможность при старте режима исполнения вывести на экран какой-либо точечный рисунок (стартовую заставку). Кнопка Выбор выводит стандартный диалог, где выбирается файл с рисунком, поле содержит путь и имя выбранного файла.
Группа Вид после старта	Кнопка Стартовая мнемосхема - при переходе в режим исполнения включается мнемосхема, которую назначили стартовой. Переход к редактированию стартовой мнемосхемы для конкретного компьютера происходит на закладке окна у свойств Компьютера. Кнопка навигатор проекта - при переходе в режим исполнения на экране отображаются страницы свойств элементов в режиме исполнения.
Группа Внешний вид при активном документе	Отмечается флажками что показывать: Основная панель (панель инструментов), Панель вызова документов, Дерево объектов, Дерево Системы, Строка статуса.
Группа Загрузка мнемосхем и окон	Флаг <i>Не выгружать из памяти при закрытии</i> – при закрытии окна оно только скрывается, но не выгружается из памяти целиком, что позволяет быстрее его отобразить при следующем открытии. Флаг <i>Загрузить все при старте</i> - при переходе в RT все окна открываются в скрытом режиме (сделано для того чтобы первое открытие было быстрым). Флаг <i>Документы открывать как окна</i> - если для мнемосхем задан режим открытия документ, то они открываются как окна без заголовка и рамки (это необходимо для использования предыдущих двух настроек). Флаг <i>Закрывать при открытии свернутого нового окна</i> - если новое окно открывается на месте существующего, предыдущее окно, в свою очередь, закрывается. У окон не должно быть заголовка.
Прятать неиспользуемые закладки	Если установлен этот флаг, в режиме исполнения не будет закладок, на которых нет элементов управления, необходимых в режиме исполнения. В противном случае будут отображаться все закладки, которые есть в режиме разработки.

Страницы свойств Компьютеров

С помощью страниц свойств **Компьютеров** настраиваются свойства конкретного компьютера.

В поле **Имя** вносится любое имя компьютера, например *АРМ Диспетчера*. Разрешение на просмотр позволяет разделить область видимости объекта для различных мест, т.е. объект может быть виден в тренде, а в интернет-клиенте – нет. Кнопка сформировать опции для заказа формирует конфигурацию необходимую для работы проекта.

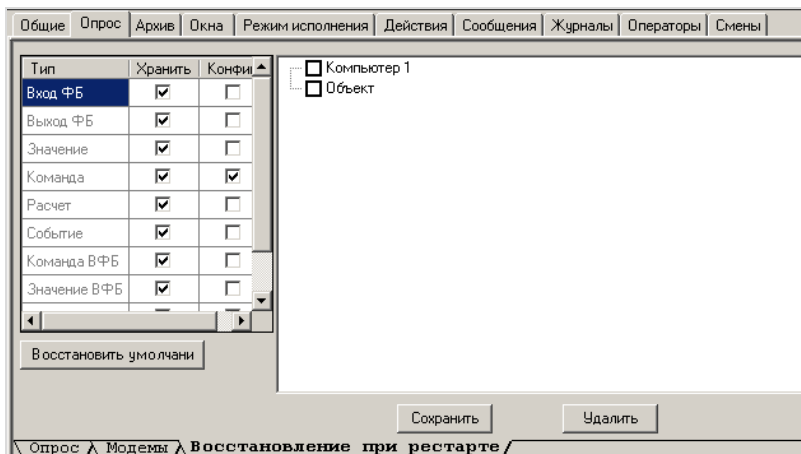
Содержание закладки **Опрос** и ее внешний вид отличается от аналогичной закладки для **Системы** только наличием кнопки **Вернуть умолчания**, которая позволит установить первоначальные настройки указанные в настройках **Системы**. На листе **Модемы** настраиваются модемы, которые могут быть подключены к компьютеру.

Элементы управления:

- Поле *Время перезагрузки* - время (с), на которое отключается GSM-модем при его перезагрузке;
- Поле *Максимальное количество попыток соединения* - максимальное количество неуспешных подряд попыток соединения с другим модемом, по исчерпанию которых необходимо произвести перезагрузку GSM-модема. При значении 0 данная функция выключена;
- Поле *Период периодической перезагрузки* - период (с) обязательной периодической перезагрузки GSM-модема. При значении 0 данная функция выключена;
- Поле *Таймаут в командном режиме* - таймаут (мс) разрыва в потоке данных при выполнении команд во время работы с модемом;

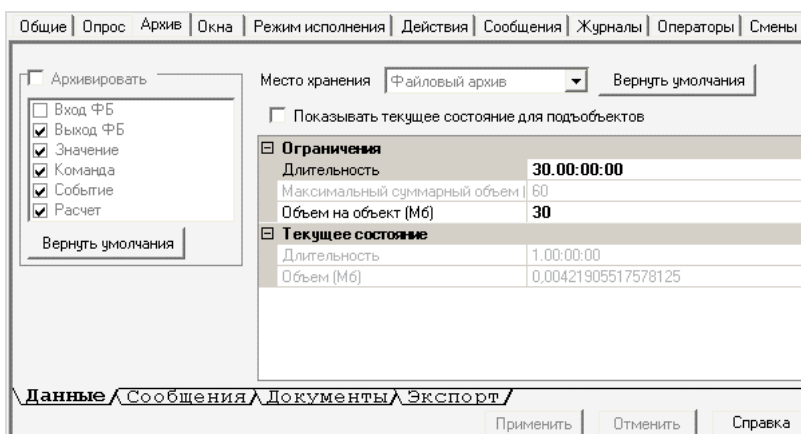
- Поле *Таймаут в режиме соединения* - таймаут (мс) разрыва в потоке данных при соединении с модемом;
- Поле *Таймаут ожидания соединения* - таймаут (с) ожидания установки соединения с другим модемом;
- Поле *Таймаут операции чтения* - время (мс) фиксирования окончания операции чтения по отсутствию принимаемых данных.

На листе **Восстановление при рестарте** настраиваются восстановления для разных типов параметров.



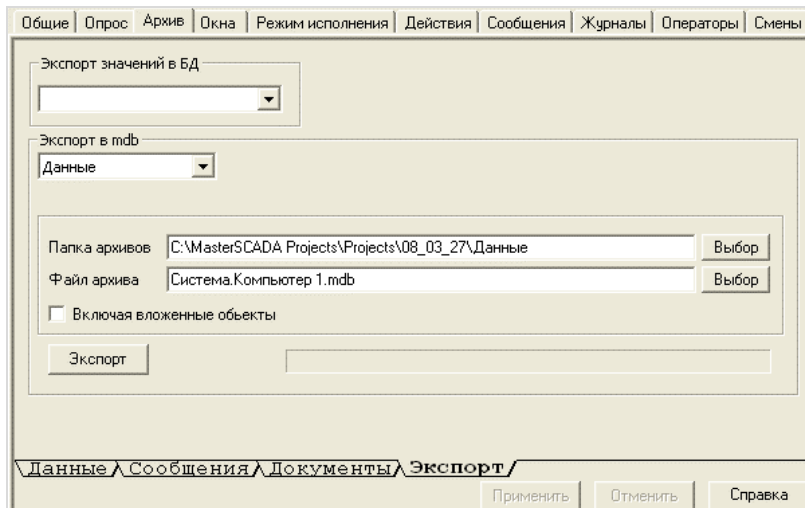
Столбец *Хранить* указывает, нужно ли хранить элементы данного типа, если задана настройка восстановление при рестарте у родительского элемента. У конкретного элемента можно будет переопределить данную настройку. Столбец *Конфигурация* указывает, что значения для восстановления нужно хранить в специальном файле. При работе в контроллере эти значения будут включены в конфигурацию контроллера или в отдельный файл values.bin.

Закладка **Архив** служит для переопределения места хранения данных по умолчанию для объектов исполняемых на компьютере. Также могут быть изменены настройки сделанные в системе, определяющие, какие элементы проекта архивируются. При нажатии на кнопку *Вернуть умолчания* задается место хранения определенное на странице свойств системы архив/данные.



На листе **Сообщения** определяется место хранения архивов сообщений для объектов исполняемых на текущем компьютере. Содержание закладки **Документы** и ее внешний вид отличается от аналогичной закладки для системы только наличием кнопки *Восстановить умолчания*.

На листе **Экспорт** осуществляется экспорт архивов данных объектов, принадлежащих данному компьютеру в формате .mdb. Экспорт данных и сообщений происходит отдельно. Все элементы, касающиеся экспорта относятся к архиву, который на данный момент выделен в таблице (данные или сообщения). При необходимости можно задать имя экспортированного файла и указать путь к нему (по умолчанию это - папка проекта). Флаг *Включая вложенные объекты* означает включение в экспортируемый архив данных из вложенных объектов всех уровней вложенности.

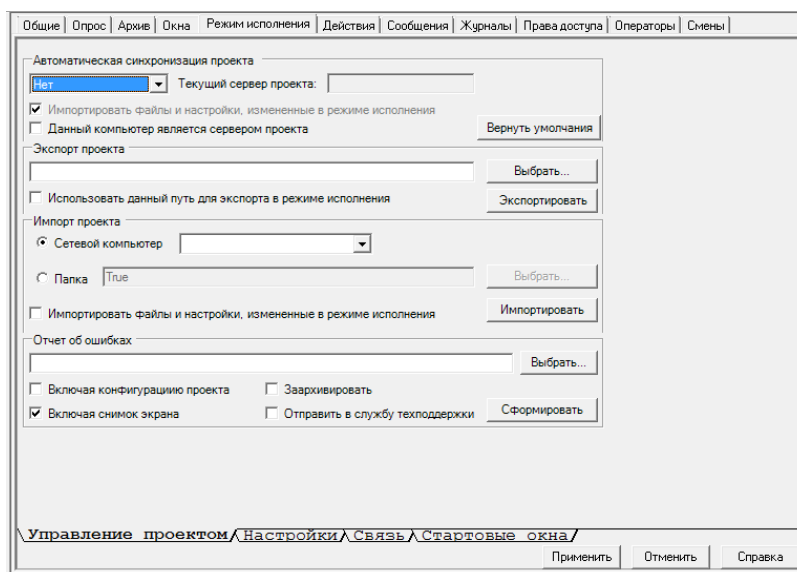


Внешний вид и содержание закладки **Окна** аналогично закладке у *Системы*. Здесь возможно изменять высоту и ширину окон документов в зависимости от параметров конкретного компьютера.

Кнопка *Стартовая мнемосхема* открывает окно редактирования стартовой мнемосхемы, с которой будет начинаться работа в режиме исполнения (если на закладке *режим исполнения* в группе *вид после старта* установлена кнопка *стартовая мнемосхема*).

Кнопка *Выбор способа открытия* устанавливает вид стартовой мнемосхемы при запуске режима исполнения (документ или окно).

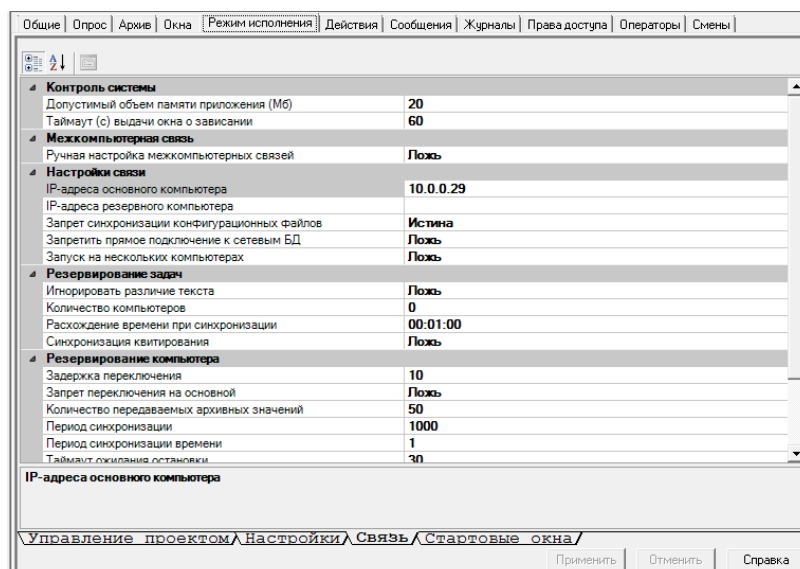
Настройки закладки **Режим исполнения** подробно описан ранее. С данной страницы можно выполнять импорт, экспорт проекта, а также в случае возникновения ошибок в работе проекта формировать отчет об ошибках masterSCADA.



Лист **Связь** на закладке *Режим исполнения* используется в случае построение систем с архитектурами: сервер-сервер, клиент-сервер, а также при резервировании компьютеров. Элементы управления выполняют следующие функции:

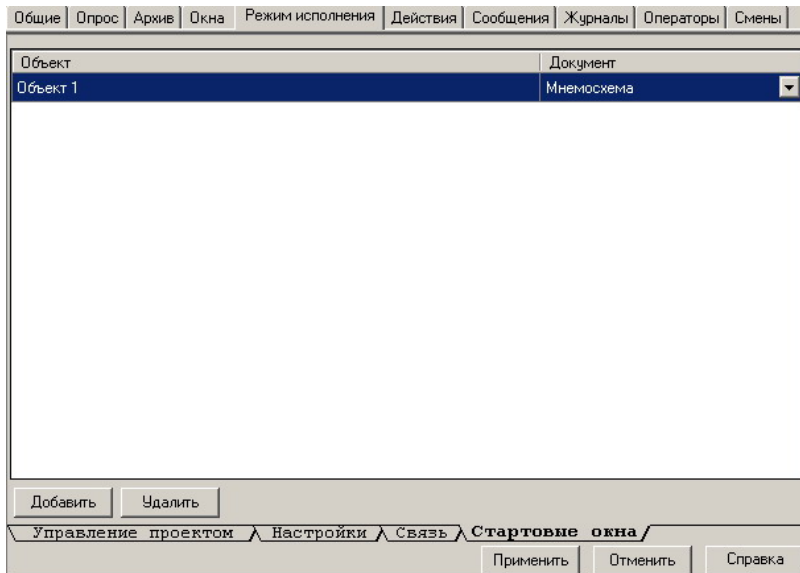
- Поле *Допустимый объем памяти приложения (Мб)* - если данный параметр не равен **0**, то при превышении заданного объема формируется дамп памяти процесса и выдается сообщение. Дамп сохраняются в папке с проектами, по пути *MasterSCADA Projects\sys_log\MemoryDumps*. Таким образом, измеряется виртуальная память процесса;
- Поле *Таймаут(с) выдача окна о зависании* - выдает окно о зависании программы, если в течении заданного времени не был получен ответ от MasterSCADA;
- Поле *Ручная настройка межкомпьютерных связей* - если в поле стоит значение *Истина*, то есть возможность в ручную выбрать с какими компьютерами необходимо инициативно установить связь, стоит значение *Ложь* - связь устанавливается автоматически;

- Поле *Запретить прямое подключение к сетевым БД* если условие *Истина*, то взаимодействие с БД происходит через запущенный на сервере экземпляр MasterSCADA, если условие *Ложь*, то запрос формируется на прямую к удаленной БД;
- Поле *Запуск на нескольких компьютерах* если условие *Истина*, то возможен запуск на нескольких физических компьютерах без добавления их в дерево системы (с указаниями одного логического имени), если условие *Ложь*, то возможен запуск только компьютеров расположенных в дереве системы. При условии *Истина* выполнять объекты на данном компьютере нельзя возможно только *выполнять на всех*;
- Поле *Расхождение времени при синхронизации* - задается только при настройке квитирования сообщений (поле синхронизация квитирования имеет значение *Истина*). Системное время компьютеров, на которых производится резервирование задач, не должны отличаться более чем на это время. В противном случае квитирование не будет передано;
- Поле *Синхронизация квитирования* - если устанавливается значение *Истина*, с одного компьютера на другой, работающий в автономном режиме, будет передаваться признак квитирования сообщений. Для этого на каждом компьютере следует запустить один и тот же проект. При квитировании сообщения на одном, на другие компьютеры, работающие в данный момент, будет пересылаться информация о его квитировании. Если компьютер был выключен, то при включении в архивном журнале это сообщение останется не квитированным. Квитировать можно только последнее активное сообщение. Информация о квитировании неактуальных сообщений (если по этому источнику появилось новое сообщение) не будет передаваться.



- Поле *Задержка переключения* - время (с), через которое произойдет переключение после отказа компьютера;
- Поле *Количество передаваемых архивных значений* - максимальное количество передаваемых архивных значений по каждому параметру за один цикл синхронизации;
- Поле *Период синхронизации* - период (мс) синхронизации резервного компьютера с основным компьютером;
- Поле *Период синхронизации времени* - период (ч) синхронизации времени резервного компьютера со временем основного компьютера;
- Поле *Таймаут ожидания остановки* - таймаут (с) ожидания остановки при перезапуске задачи резервирования.

На листе *Стартовые окна* определяется список документов компьютера, которые загрузятся при старте режима исполнения MasterSCADA.



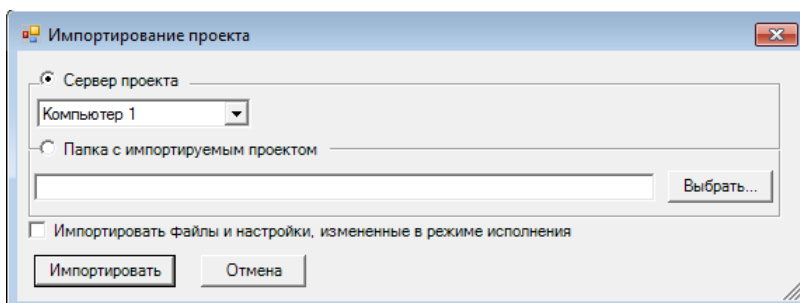
Необходимо перетащить объект, документ которого необходимо открывать, на страницу свойств, а затем из выпадающего списка выбрать тип документа, который должен загрузиться при старте. Если необходимо при старте загружать только мнемосхему, какого-либо объекта или компьютера, то в этом случае, достаточно сделать настройку на странице свойств *Окна* объекта или компьютера.

Автоматическая синхронизация проекта предназначена для быстрого внесения изменений в проекты сетевой архитектуры. Один из компьютеров назначается сервером проекта – проект, исполняемый на нем, считается эталонным, именно на этот компьютер разработчик должен вносить изменения (например, через импорт папки с проектом). Если остальные компьютеры сетевого проекта (клиенты) обнаруживают, что их проект отличается от проекта на сервере, то происходит загрузка свежего проекта. Для обновления клиента должны быть выполнены следующие условия:

- На сервере должен быть запущен проект в RT;
- Клиент и сервер должны находится в одной подсети (проверить это можно с помощью команды *ping адрес сервера*).

Обновление проекта с сетевого компьютера возможно в двух режимах - ручном и автоматическом:

- **Для ручного обновления.** В режиме разработки нужно выполнить команду *ПроектИмпортировать....* В диалог импорта добавлена возможность выбора сетевого компьютера проекта с которого будет произведена загрузка проекта. На этом компьютере должен быть запущен данный проект. В режиме исполнения можно выполнить команду *СервисИмпортировать...* или выполнить обновление с закладки компьютера *Режим исполнения - Управление проектом* (при этом проект на клиенте перезапускается после обновления.);



- **Для автоматического обновления** у одного компьютера нужно включить опцию *Является сервером проекта*. При запуске клиента идет сравнение версии проекта клиента с версией запущенной на сервере. Если проекты отличаются, то выполняется импорт проекта с сервера (при этом проект на клиенте перезапускается после обновления). Процедура происходит только при запуске проекта с командной строки в режиме исполнения (если запускать из режиме разработки, то автоматического обновления не происходит, нужно обновлять вручную). Флаг **Импортировать файлы**

и настройки, измененные в режиме исполнения играет ту же роль, что в диалоге импорта, если включен то копируются файлы, изменяемые в режиме исполнения (файл горячего рестарта, режима исполнения, настройки тренда и т.п.).

Закладка **Действия** содержит таблицу приблизительно такого вида:

Время	Оператор	Компьютер	Действие
12:06:01	Иванов	Компьютер 1	Вход в систему
12:06:33	Иванов	Компьютер 1	Выход из системь

В таблицу заносится информация о действиях оператора в режиме исполнения. Какие именно действия оператора фиксировать устанавливается при настройке должностей. Отображаются действия только с системой или с элементами текущего компьютера.

Содержание закладки **Сообщения** и ее внешний вид отличается от аналогичной закладки для *Системы* только наличием кнопки *Вернуть умолчания*.

Закладка **Журналы** содержит перечень доступных журналов. Для каждого журнала можно настроить способ открытия, а также расположение на экране. Столбец *Название* - имя журнала. Столбец *Тип* - с помощью выпадающего списка для каждого журнала можно настроить способ его открытия. Столбец *Свойства* - при нажатии на кнопки, находящиеся в ячейках этого столбца открывается диалоговое окно, с помощью которого для каждого тренда можно настроить параметры окна (расположение на экране, а также его размеры)

Название	Тип	Свойства
Основной журнал	Документ	...
Неквитированные сообщен	Документ	...
Активные сообщения	Документ	...

Открыть

Удалить

Открыть сохраненные

Вернуть умолчания

После того как на закладке **Операторы** системы созданы должности операторов, на закладке **Операторы** компьютера с помощью кнопок *Добавить* и *Удалить* на должности назначаются конкретные лица (операторы). Для каждого оператора задается пароль, который он должен будет ввести, заступая в свою смену.

Должность	Оператор
Просмотр	Пушкин
	Лермонтов
	Лазртский
Оператор	Регин
	Шпкин
	Левитан
Старший оператор	Чайковский
	Мусоргский
	Прокофьев
Начальник смены	С.....

Добавить Удалить Сменить пароль

На закладке **Смены** можно проделать следующее:

- Перенастроить смены применительно к данному компьютеру;
- Назначить операторов на смены.

	Название	Начало	Конец	Должность	Оператор
1	Смена 1	8 : 00	16 : 00	Просмотр	
2	Смена 2	16 : 00	0 : 00	Оператор	
3	Смена 3	0 : 00	8 : 00	Старший оператор	
				Начальник смены	
				Начальник цеха	
				Диспетчер	
				Разработчик	sa <input checked="" type="checkbox"/>

Добавить Удалить Вернуть умолчания

Страницы свойств OPC серверов

OPC DA сервер как элемент дерева системы имеет свойства и настройки во многом схожие с настройками функциональных блоков. Поэтому ряд закладок на страницах свойств у них полностью совпадает ввиду идентичной функциональности. Это закладки - *Общие*, *Сообщения*, *Свойства*. Закладка *Опрос* отличается от аналогичной закладки функционального блока, так как настройки опроса предназначены для конфигурирования обмена данными с OPC DA сервером, и поэтому их функциональность ограничена возможностями стандарта OPC.

Имя: InSAT_DebugServer_MasterOPC_Data Server Тип: InSAT.DebugServer.MasterOPC.Data Server

Подсчитать количество связей

С OPC переменными

Разрешение на просмотр в:

тренде журнале
 интернет окне

Вернуть умолчания

Расположение OPC сервера при разработке: OCTOBER02 Сменить...

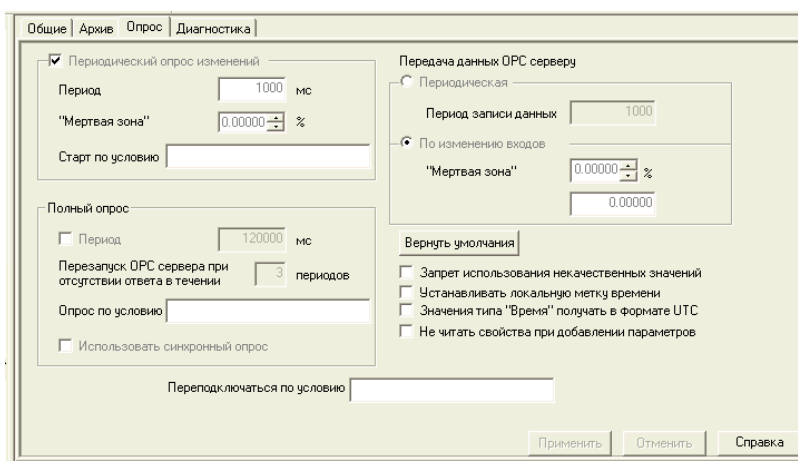
Удаленный OPC сервер в режиме исполнения

CLSID: {DEB1DE9A-5C89-468D-A74F-1010C3F0B516} Путь: JANUARY022

Функциональные возможности элементов управления:

- В поле *Имя* можно изменить стандартное имя;
- Поле *Расположение OPC сервера при разработке* указывает имя компьютера (сетевое), на котором располагается, используемый при создании проекта как источник информации о списке переменных, задействованных в обмене данными с внешними устройствами;
- Кнопка *Сменить* - при нажатии на эту кнопку появляется список доступных в данный момент компьютеров сети. Если нажать на знак + слева от имени компьютера, то откроется список установленных на нем OPC-серверов. Выбор любого OPC-сервера приведет к выбору компьютера, на котором он расположен, в качестве базового для поиска в режиме разработки проекта настраиваемого OPC-сервера. В режиме исполнения это имя компьютера не используется. MasterSCADA обращается к тому компьютеру, которому данный OPC-сервер принадлежит в дереве системы;
- Установка флага *Удаленный OPC-сервер в режиме исполнения* позволяет указать сетевое имя компьютера, на котором (в режиме исполнения) будет запускаться данный OPC-сервер. Если флаг не установлен, MasterSCADA, будет пытаться запустить OPC-сервер на локальном компьютере.

Закладка **Опрос** выглядит следующим образом:



Функциональные возможности элементов управления:

- Установка флага *Периодический опрос* означает, что через промежутки времени, указанные в поле *Период* OPC- сервер должен присылать новые данные OPC переменных в MasterSCADA, если они изменились более чем на значение, указанное в поле *Мертвая зона*;
- Поле *Период* - наследуется из свойств компьютера - закладка *Опрос*;
- Поле *Старт по условию* - в это поле можно перетащить дискретную переменную с помощью которой в режиме исполнения указывается проводить периодический опрос изменений или нет. В режиме исполнения периодический опрос проводится, если значение переменной, указанной в поле *Старт по условию* равно *Истина*, и не проводится, если *Ложь*. Переменная, с помощью которой осуществляется управление, должна принадлежать тому же компьютеру, что и OPC-сервер, иначе данные передаваться не будут. Переменная должна быть логической;
- *Полный опрос* - опрос текущих значений всех OPC переменных (вне зависимости от того, изменялись они или нет). Также будет производиться попытка переподключения к переменным, с которыми по каким-либо причинам обнаружился обрыв связи;
- Поле *Период* - период полного опроса (синхронного или асинхронного);
- Флаг *Использовать синхронный опрос* (по умолчанию всегда снят): если установлен, используется синхронный опрос, если не установлен - асинхронный;
- Поле *Перезапуск OPC-сервера при отсутствии ответа в течении ___ периодов* - указывается количество периодов опроса, после истечения которых производится перезапуск OPC-сервера в случае отсутствия ответа от OPC- сервера при асинхронном опросе. В случае синхронного опроса MasterSCADA получает данные переменных из OPC-сервера сразу. В случае асинхронного опроса с указанным периодом дается запрос OPC серверу передать данные в MasterSCADA. Если при этом ответ OPC-сервера, содержащий значения OPC переменных, не приходит в течение времени, равному числу периодов, указанному в поле *Перезапуск OPC-сервера в течение...*, то производится перезапуск OPC-сервера;
- Поле *Опрос по условию* - предназначено для перетаскивания в него переменной, с помощью которой в режиме исполнения задается необходимость проведения полного опроса. В режиме исполнения, полный опрос проводится с периодом, указанным в поле *Период*, а также, если фиксируется переход управляющей переменной из *Ложь* в *Истина* (передний фронт).
- Группа *Передача данных OPC-серверу* организует передачу данных от MasterSCADA к OPC-серверу контроллера:
 - *Периодическая* - передача данных будет осуществляться через равные промежутки времени;
 - Поле *Период* - можно изменить значение этого промежутка времени (изначально унаследованного из свойств компьютера - закладка *Опрос*);
 - *По изменению входов* - данные будут передаваться OPC-серверу как только изменятся значения на входах OPC-сервера (элемента дерева системы в MasterSCADA);
- Установка флага *Запретить использование некачественных значений* позволяет запретить использование недостоверных данных, присылаемых OPC-сервером (данные считаются недостоверными, если они имеют *плохой* признак качества);

- Флаг *Не читать свойства при добавлении параметров* - не читать описание OPC переменной (свойство переносится в поле *Описание* на закладке *общие*). При чтении данного свойства некоторые OPC серверы могут подвисать;
- Поле *Переподключаться по условию* предназначено для перетаскивания в него переменной, с помощью которой в режиме исполнения осуществится отключение от OPC-сервера и повторное подключение. Если у OPC-сервера нет других клиентов, то по стандарту OPC, в момент отписки MasterSCADA от OPC-сервера, сервер должен завершить свой процесс;
- Если установлен флаг *Устанавливать локальную метку времени*, то приходящие значения получают текущее время, а не время с OPC-сервера;
- Если установлен флаг *Значения типа Время получать в UTC*, то значения будут передаваться в формате OPC (работает не на всех типах OPC-серверов).

Некоторые Функциональные блоки и **OPC серверы** могут иметь какое-либо **Свойство** (одно или несколько), такое как, например, количество входов. Конкретные свойства Функционального блока или OPC сервера зависят от его разработчика. Информацию о свойствах какого-либо Функционального блока или OPC сервера нужно смотреть в документации к нему. Такие свойства как правило можно настраивать. На этой странице можно изменить его значение и это изменение будет соответствующим образом отображено в дереве функционального блока после нажатия кнопки *Применить*.

Свойство	Тип свойства	Значение
property NInput	Целое	5

В случае, когда разработчик не считал необходимым снабдить блок настраиваемыми свойствами таблица будет пуста.

Закладка **Опрос OPC HDA** сервера отличается от аналогичной закладки OPC DA серверов следующими полями: *Период*, которое определяет полный цикл пересчета всех элементов, *Приоритет (0-255)* - если задавать значения > 100, то эти задачи будут получать больше процессорного времени в случае его нехватки. Также более точно будет выдерживаться период цикла

Закладка **Настройки** выглядит следующим образом:

В поле *OPC-переменные* следует отметить флажком те переменные или целые группы, которые предполагается использовать в проекте. После этого в дереве системы будут сразу отображены выбранные группы и переменные, с которыми можно совершать те же действия, что и с входами и выходами функциональных блоков, то есть перетаскивать мышью для установления связей, изменять параметры страницы свойств и т.д.

В поле ввода *Расположение OPC сервера при разработке* задается имя компьютера, на котором установлен OPC HDA сервер при разработке. В случае, если OPC HDA сервер и разрабатываемый проект находятся на одном компьютере, то в этом поле стоит значение *localhost*. Кнопка *Подключиться* устанавливает связь с компьютером, на котором находится OPC HDA сервер в режиме разработки. Используется в случае изменения данных в поле ввода *Расположение OPC сервера при разработке*.

Поле *Интервал агрегатных значений (с)* - интервал времени, с которым происходит обработка параметра агрегатной функцией (агрегаты – это набор функций обработки параметра количество записей, минимум, максимум и т.д.). При подключении OPC-клиент запрашивает поддерживаемые агрегаты-функции.

Поле *Период Запроса* устанавливает желаемый период времени (в днях), за который OPC будет запрашивать данные у сервера при запуске. Если архивы в MasterSCADA существуют, то данная настройка не используется, если архивных записей нет, то необходимо указать период запроса. OPC HDA клиент может работать в трех режимах (поле *Получение данных*):

- *Подписка* - клиент при старте подписывается получать новые значения;
- *Чтение и подписка* - клиент при старте читает, все значения, которых у него нет. Когда чтение закончено он подписывается на новые значения;
- *Чтение* - клиент с периодом с закладки *Опрос* читает значения из сервера.

Поле *Время ожидания ответа* задает сколько периодов получения данных MasterSCADA ждет ответа от сервера на запрос перед тем как послать запрос заново.

Страницы свойств коннекторов к базам данных

Страницы свойств БД-коннекторов позволяют произвести настройку БД-коннектора. БД-коннектор, как элемент дерева системы имеет следующие страницы свойств: *Общие*, *Опрос*, *Параметры*, *Экспорт архивов*.

Настройки, производимые на странице **Опрос** в основном предназначены для наследования подчиненными элементами - *Блоками процедур*. Флаг *Периодический* предназначен для наследования подчиненными элементами. Для *Блоков процедур*, данного БД-коннектора задает периодический способ их пересчета. Полный цикл пересчета всех блоков процедур задается в поле *Период*. Если нет необходимости пересчитывать Блоки процедур, входящие в данный БД-коннектор каждый цикл (такт), то можно установить кратность пересчета в окне *Каждый ____ такт*. Если задавать значения > 100 в поле *Приоритет 0-255*, то эти задачи будут получать больше процессорного времени в случае его нехватки. Также более точно будет выдерживаться период цикла.

Флаг *По изменению входов* предназначен для наследования подчиненными элементами. Для блоков процедур, входящих в состав данного БД-коннектора задает способ их пересчета - по изменению входов. Как только изменение данных на входах блока процедур превысит величину мертвой зоны, блоку процедур присваивается особое свойство как нуждающемуся в пересчете и как только подойдет его очередь, он немедленно будет пересчитан. С некоторой погрешностью можно считать, что это происходит сразу после обновления данных на входах.

Остальные элементы управления выполняют аналогичный функционал описанный в предыдущих разделах.

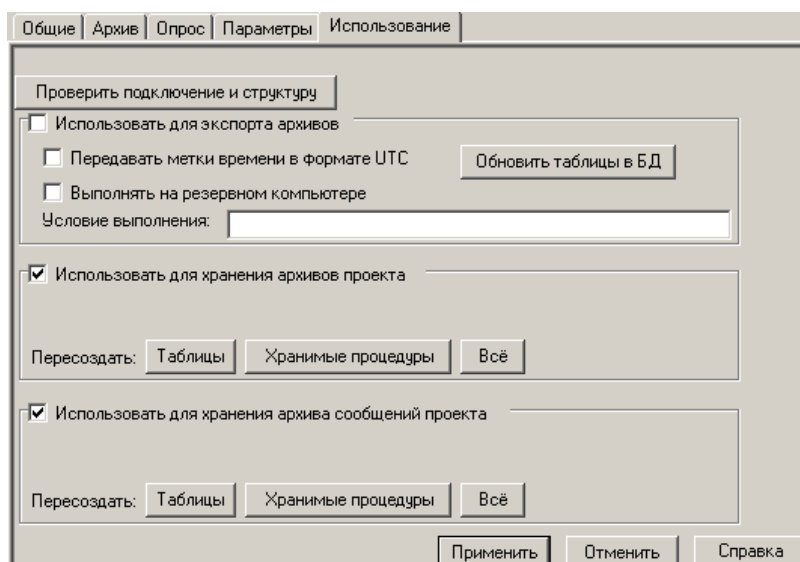
Закладка **Параметры** служит для настройки параметров соединения с базой данных. Эти закладки различны и их наполнение зависит от выбранной базы данных.

MS SQL Server. В полях *Сервер* и *База данных* нужно задать соответственно имена сервера, базу данных с которого планируется использовать, и имя самой базы данных. Группа *Подключение* позволяет сконфигурировать режим доступа к SQL Server, используя либо аутентификацию Windows, либо аутентификацию SQL Server. В поле *Сервер* можно указать несколько ip-адресов (также как в сетевом взаимодействии). Разделитель ";" - отделяет ip, которые участвуют в переборе. Разделитель ":" - отделяет ip, которые не участвуют в переборе. В режиме разработки используется только первый ip. Если произошла ошибка подключения, то, при следующей попытке подключения будет использоваться следующий ip, кроме отделенных ":".

Oracle. Необходимо задать имя Базы данных, а также имя пользователя и пароль.

InterBase. Для работы с данной базой данных используется провайдер IBProviderFree. Необходимо задать имя файла данных, а также имя пользователя и пароль.

На закладке **Использование** указывается, будет ли система использовать для экспорта и хранения архивов данных БД-коннектор.



Функциональные возможности элементов управления:

- Флаг *Использовать для экспорта архивов* разрешает экспорт архивов в данную БД;
- Флаг *Выполнять на резервном компьютере* - на резервном компьютере экспорт ведется параллельно основному (имеет смысл, если БД также резервируется и расположена на том же компьютере, где стоит MasterSCADA);
- В поле *Условие выполнения* можно задать логическую переменную. Экспорт будет выполняться только, когда она *Истина*;
- Кнопка *Обновить таблицы в БД* позволяет в режиме разработки создать таблицы/процедуры (если они не были созданы ранее), а также занести в таблицу *Items* список экспортируемых параметров (эта же процедура выполняется автоматически при запуске режима исполнения);
- Флаг *Использовать для хранения архивов проекта* разрешает хранение архивов в данной БД;
- Кнопка *Пересоздать таблицы* создает/обновляет таблицы в БД (которые используются для хранения архива данных). Если в БД уже были эти таблицы, структура которых в текущей версии MasterSCADA изменилась, они удаляются вместе со всем содержимым;
- Кнопка доступна только в режиме разработки. *Пересоздать хранимые процедуры* и *Пересоздать всё* работают аналогично;
- Флаг *Использовать для хранения архива сообщений проекта* разрешает хранение архивов в данной БД. Кнопки пересоздания работают аналогично;
- При нажатии кнопки *Проверить подключение и структуру*, происходит проверка наличия необходимых таблиц/процедур для ведения архивов данных и сообщений. Помимо наличия, проверяется соответствие версий таблиц/процедур. Если найдены несоответствия, то в соответствующих группах выдаются сообщения: *Структура таблиц не соответствует версии MasterSCADA*, *Код хранимых процедур не соответствует версии MasterSCADA*. Кнопка доступна в режиме разработки и в режиме исполнения.

Логика создания служебных таблиц/хранимых процедур:

- *Экспорт архивов*. При старте проекта в этой БД будут созданы таблицы и служебные хранимые процедуры, если в БД отсутствует таблица свойства;
- *Хранение архивов*. При старте проекта в этой БД будут созданы таблицы и служебные хранимые процедуры, если они в БД отсутствуют (наличие проверяется по каждой таблице/процедуре индивидуально). Если таблицы/процедуры есть, но их структура которых в текущей версии MasterSCADA изменилась, то в протокол работы записывается сообщение *Структура базы не соответствует текущей версии MasterSCADA (тип несоответствия)*. В этом случае необходимо вручную выполнить обновление БД (кнопками *Обновить* в соответствующих группах).

Страницы свойств Контроллеров

Элементы управления на страницах **Общие**, **Опрос** работают аналогично элементам описанным ранее.

На закладке **Архив** возможно настроить некоторые свойства контроллера. Поле *Архивировать* определяет умолчания для дочерних элементов, значения каких элементов проекта будут архивироваться. В дальнейшем для каждого отдельного элемента деревьев настройку можно переопределить. Выпадающий список *Архив в контроллере* появляется на закладке архив в том случае, если в настройках контроллера для параметра *Количество дополнительных задач архивирования* установлена одна или более дополнительных задач. После чего можно выбрать архив, где будут храниться параметры контроллера.

Закладка свойств контроллера **Настройка** позволяет выполнять следующие настройки. Поле *Ограничение на глубину чтения из архива (0-не ограничивать)* - читаются только архивные записи старше этого интервала. Более старые игнорируются (ограничение действует и при начале опроса и в его ходе). Поле *Размер архива сообщений в памяти контроллера, kB* - объем архива сообщений в памяти контроллера. При достижении этого объема, перед записью новых сообщений, из архива удаляются наиболее старые записи.

Поле *Контролировать мертвую зону в контроллере* - если установлено значение *Истина*, то в контроллере будет учитываться значение мертвой зоны для переменных. Если в поле *Хранить настройки архивирования в отдельном файле* стоит значение *Истина*, то значения и настройки архивирования хранятся в отдельном файле *archives.bin*. Настройки включают в себя список архивируемых переменных и их мертвые зоны. Конфигурация контроллера при этом не меняется.

Поле *Максимальное расхождение времени в архиве* - в случае записи в архив контроллера немономонных архивов (например, драйверами устройств) позволяет задать максимальный интервал между такими записями.

Если в поле *Выполнять архивирование в контроллере* установлено значение *Истина*, то в контроллере будет выполняться архивирование тех переменных, для которых установлен флаг *Архивировать* на странице свойств *Общие*. При значении *Истина* становятся доступны остальные поля этой группы, а также в дереве системы появляется переменная *сохранение данных*. Если в поле установлено значение *Ложь*, то архивирование тех переменных, для которых установлен флаг *Архивировать*, будет выполняться только на компьютере.

Общие Архив Опрос Настройки	
Архивирование	
Ограничение на глубину чтения из архива (0 - не ограничивать)	00:00:00
Размер архива сообщений в памяти контроллера, kB	0
Контролировать мертвую зону в контроллере	Истина
Хранить настройки архивирования в отдельном файле	Истина
Максимальное расхождение времен в архиве	00:00:00
Выполнять архивирование в контроллере	Истина
Количество дополнительных задач архивирования	0
Максимальное количество параметров (0 - не ограничено)	100
Использовать архивные значения как текущие	Истина
Размер архива данных в памяти контроллера, kB	64
Период периодической записи значений (с)	0
Горячий рестарт	
Использовать горячий рестарт	Истина
Место хранения данных	
Период сохранения данных (с)	1
Время актуальности данных (с), 0 - всегда	0
Конфигурация проекта	
Разрешить опрос при несоответствии конфигураций	Истина
Игнорировать различие версий при горячем рестарте или резервировании	Истина
Запрет автоматической загрузки конфигурации	Истина
Адреса, с которых разрешена загрузка конфигурации	
Прямой опрос с сетевых компьютеров	Истина
Включение значений хранимых команд в конфигурацию	Истина
Хранить настройки контроллера в отдельном файле	Истина
Межконтроллерный обмен	
Период межконтроллерного обмена	1000
Приоритет межконтроллерного обмена	100
Протоколы	
Основные порты для протоколов	
Использовать GSM модем	Истина
Разное	
Использовать Ethernet	Истина
Тип	ADAM-5510
Количество последовательных портов	3
Вещественные числа двойной точности	Истина
Резервирование	
Использовать резервирование	Истина
Период резервирования	1000
Синхронизация времени	
Автоматическая синхронизация	Полная
Адреса, с которых разрешена синхронизация	
Период синхронизации (с)	3600
Чтение	
Приоритет параметров открытых окон	0
Период приоритетного опроса	100
Отображение значений в режиме исполнения	Истина
Период опроса	1000
Максимальный период опроса	30000
<input type="button" value="Применить"/> <input type="button" value="Отменить"/> <input type="button" value="Справка"/>	

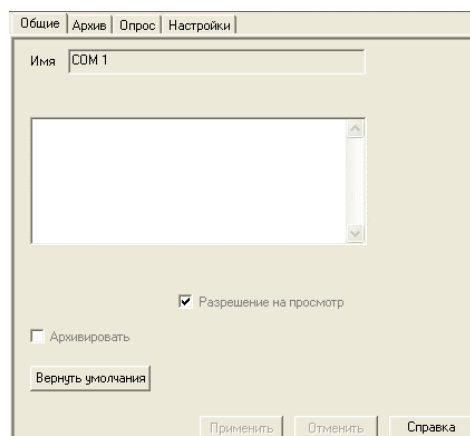
Функциональные возможности элементов управления:

- В поле *Количество дополнительных задач архивирования* задается количество задач. По умолчанию задачи именуются *Дополнительный архив N*. Для каждой задачи можно задать имя и максимальный объем в памяти контроллера;
- Поле *Использовать архивные значения как текущие* - не читать отдельно текущие значения архивируемых параметров. Использовать последние архивные значения как текущие;
- Поле *Размер архива в памяти контроллера, kB* - объем архива данных в памяти контроллера. При достижении этого объема, перед записью новых данных, из архива удаляются наиболее старые записи;
- Поле *Период периодической записи значений* - длительность хранения архивных данных. Если срабатывает это ограничение, то перед записью новых данных, из архива удаляются наиболее старые записи;
- Если в поле *Использовать горячий рестарт* установлено значение *Истина*, то после аварийного останова контроллера и последующего рестарта в нем будут восстановлены актуальные данные переменных и информация о состоянии ФБ, для которых установлен флаг *Восстановление при рестарте* на странице свойств *Опрос*. Также при значении *Истина* становятся доступны остальные поля этой группы;
- Поле *Место хранения данных* - место на диске контроллера, куда сохраняются данные, которые требуется восстанавливать при рестарте. Если не указано, использует-

- ся стандартное место для данного контроллера (если в контроллере присутствует модуль энергонезависимой памяти SRAM, то запись ведется в него, если нет, то в папку, где расположен исполняемый файл MasterPLC);
- Поле *Период сохранения данных (с)* - период, с которым сохраняются данные переменных и состояние функциональных блоков. Если в контроллере не установлена память SRAM, а используется Flash-память, то применять эту функцию не рекомендуется. В этом случае в поле нужно установить значение 0, а сохранение производить при помощи переменной Сохранение данных(See 6.10) в дереве Системы. При аварийном останове контроллера на вход этой переменной должен поступать сигнал от источника бесперебойного питания о том, что он включился;
 - Поле *Время актуальности данных (с), 0 - всегда* - интервал времени, отсчитываемый с момента последнего сохранения, в течение которого сохраненные данные считаются актуальными. Если с момента последнего сохранения данных прошло больше времени, чем указано, то они восстановлены не будут. При значении 0 сохраненные данные всегда считаются актуальными и восстанавливаются при рестарте;
 - Поле *Разрешить опрос при несоответствии конфигураций* - если конфигурации не совпадают, и если в поле установлено значение *Истина*, то опрос контроллера будет происходить. Если значение будет установлено *Ложь*, то при различных конфигурациях опрос происходить не будет;
 - Поле *Игнорировать различие версий при горячем рестарте* - если установлено значение *Истина* и используется горячий рестарт, то при загрузке в контроллер новой конфигурации будут восстановлены значения из горячего рестарта;
 - Если в поле *Запрет автоматической загрузки конфигурации* стоит значение *Ложь*, то при старте режима исполнения сравниваются конфигурация контроллера и компьютера, а также файлы настроек и настроек архивирования, в случае, если эти компоненты отличаются, то в контроллер загружаются версии компьютера;
 - Поле *Адреса, с которых разрешена загрузка конфигурации* - IP адреса компьютеров, которые могут выполнять загрузку конфигурации. IP адреса вводятся через запятую. Если поле пусто, то выполняет загрузку только тот компьютер, в который вставлен контроллер. Если IP адреса не заданы в данном поле, но стоит настройка *Прямой опрос с сетевых компьютеров*, то выполнять загрузку может любой компьютер проекта;
 - Поле *Прямой опрос с сетевых компьютеров* - если установлено значение *Истина*, то контроллер могут опрашивать несколько компьютеров проекта, объекты которых связаны с переменными контроллера;
 - Поле *Включение значений хранимых команд в конфигурацию* - если контроллер не поддерживает отдельный файл конфигурации (не имеет полноценной файловой системы), то установка этого флага позволит в состав конфигурации проекта включать хранимые команды;
 - Если в поле *Хранить настройки контроллера в отдельном файле* стоит значение *Истина*, то настройки контроллера будут храниться в отдельном файле (*values.bin*). Под настройками контроллера понимается: значения команд, у которых стоит флаг *Восстанавливать при рестарте*, а также настройки ФБ контроллера, измененные в режиме исполнения;
 - Поле *Период межконтроллерного обмена* - период обмена данными между контроллерами;
 - Поле *Приоритет межконтроллерного обмена* - если установлено значение больше 100, например, 101, то задача передачи данных между контроллерами имеет более высокий приоритет, по сравнению с другими задачами, которые выполняются в данном контроллере;
 - Поле *Основные порты для протоколов* (связь с верхним уровнем, связь с резервным контроллером, связь с другими контроллерами). Настраивается, если несколько портов настроены на работу по одинаковым направлениям (например, каждый порт настроен и на связь с верхним уровнем, и на работу с резервным контроллером), в случае если исправны оба порта, то указывается по какому порту будет проходить тот или иной канал по умолчанию. В случае, если один из портов выйдет из строя, то по одному порту пойдут все потоки;
 - Если в поле *Использовать GSM модем* установлено значение *Истина*, то контроллеру будет доступен модем, подключенный к его COM-порту и в дереве системы у контроллера появится группа переменных модем;
 - Поле *Использовать Ethernet* для некоторых контроллеров (в которых по умолчанию нет порта Ethernet) можно устанавливать этот порт;

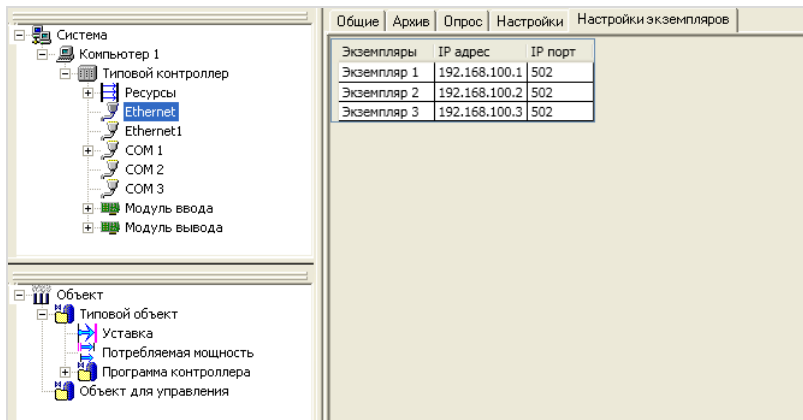
- В поле *Количество последовательных портов* задается количество доступных последовательных портов контроллера. (применяется, если используются расширительные корзины);
- Поле *Вещественные числа двойной точности* – настройка определяется исполнительной системой MasterPLC, которая загружается в контроллер. Это свойство задается автоматически при переходе в режим исполнения;
- Если в поле *Использовать резервирование* установлено значение *Истина*, то для управления процессом используются два одинаковых контроллера. При значении *Истина* в дереве системы появляется группа переменных *Резервирование*;
- Поле *Период резервирования* - период, с которым передаются данные от основного контроллера резервному;
- Поле *Автоматическая синхронизация*:
 - *Полная* - системное время в контроллере устанавливается равным времени в компьютере;
 - *Для архивов* - системное время в контроллере не меняется, но контроллер хранит смещение от времени компьютера для записи архивов;
 - *Не выполнять*;
- Поле *Адреса, с которых разрешена синхронизация* - IP адреса компьютеров, которые могут выполнять автоматическую синхронизацию. Если не задано, синхронизирует время только компьютер, в который вставлен контроллер;
- Поле *Период синхронизации, (с)* - время, с которым начинается синхронизация времени. Если время контроллера отлично от времени компьютера, то оно будет изменено на время компьютера;
- Поле *Период приоритетного опроса* - период, с которым будут передаваться приоритетные параметры на верхний уровень. Параметр считается приоритетным, если у него установлен флаг *Приоритетный параметр* на странице свойств *Опрос выхода*, а также параметры, которые отображаются в окнах управления исполнительными механизмами (в случае, если окно открыто);
- Если в поле *Отображение значений в режиме исполнения* установлено значение *Истина*, то в режиме исполнения значения переменных будут отображаться в дереве объектов и на схеме ФБ;
- Поле *Период опроса* - время между началами циклов чтения. Если текущий цикл чтения завершен и с момента начала последнего цикла чтения прошло больше времени, то начинается новый цикл;
- Поле *Максимальный период опроса* - период принудительного чтения. Если текущий цикл чтения не завершен и с момента начала текущего цикла чтения прошло больше времени, то включается режим принудительного чтения. В этом режиме команды чтения по приоритету становятся выше команд записи и отладки. При значении 0 данная функция выключена. Принудительное чтение прерывает чтение архива.

Страница свойств **Портов** предоставлена ниже.



Флаг *Архивировать* предназначен для наследования всеми элементами, принадлежащими данному контроллеру, остальные элементы управления аналогичны описанным ранее.

На странице *Настройки экземпляров(Ethernet)* производится настройка связи экземпляров контроллеров.



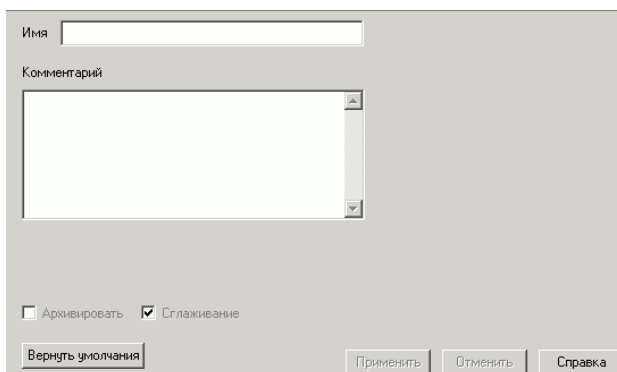
В таблице каждому экземпляру контроллера присваивается IP адрес и порт для связи с реальным контроллером. В случае использования Windows-контроллера, нужно не задавать поле IP адреса, а указать разные порты. В этом случае для каждого экземпляра будет запущена отдельная копия эмулятора.

Страницы свойств Модулей ввода-вывода

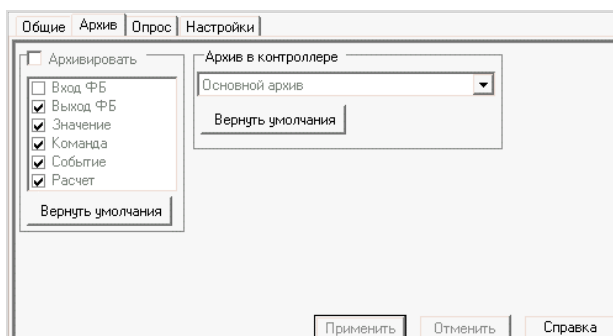
Некоторые дискретные входные сигналы модулей могут быть использованы как счетчики. Для этого необходимо установить флаг в поле *Использовать как счетчик*. При этом вместо одного дискретного входного сигнала появится группа из двух сигналов:

- CI - целочисленное значение счетчика;
- Reset - выходной сигнал для сброса.

Флаг *Сглаживание* служит для наследования подчиненными элементами. Определяет способ записи данных в архив. Остальные элементы управления аналогичны элементам контроллера.

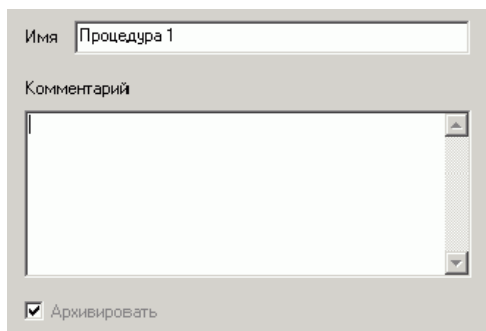


На закладке **Архив** возможно настроить Выпадающий список *Архив в контроллере* с помощью которого выбирается архив, где будут храниться параметры данного элемента. Данное поле появляется на закладке архив в том случае, если в настройках контроллера для параметра *Количество дополнительных задач архивирования* установлена одна или более дополнительных задач.

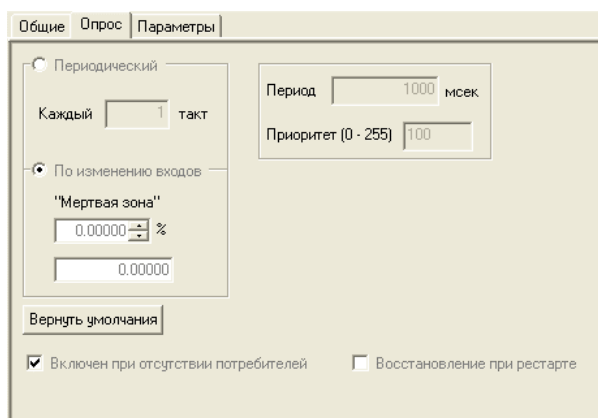


Элементы из закладки **Опрос** аналогичны элементам контроллера.

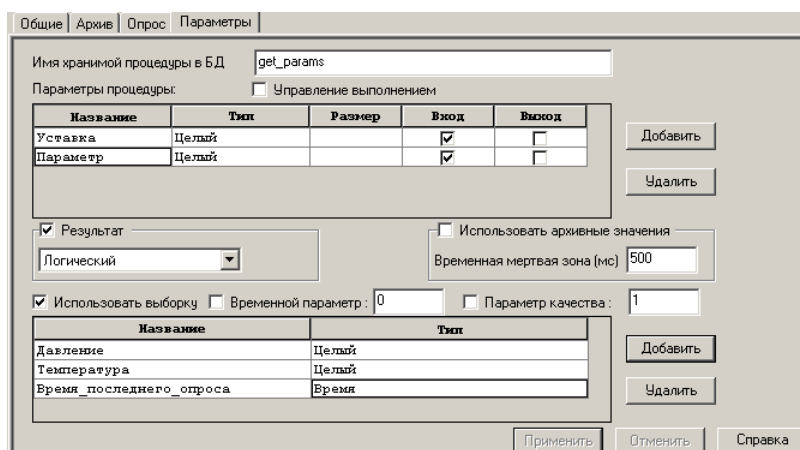
Страницы свойств Блоков процедур и Блоков функций



Поле *Имя* позволяет задать требуемое имя *Блока процедуры*. Реальное имя хранимой процедуры базы данных задается на закладке *Параметры* блока процедуры. Это позволяет создать несколько блоков процедур (с разными именами), ссылающихся на одну и ту же хранимую процедуру в базе данных. Элементы управления закладки опрос аналогичны элементам контроллера.



На закладке *Параметры* блока процедуры настраиваются следующие параметры.



В поле *Имя хранимой процедуры в БД* задается реальное имя хранимой процедуры в базе данных. Несколько блоков процедур могут обращаться к одной хранимой процедуре в БД. Если имя блока процедур в дереве не было изменено явно, то оно будет соответствовать имени хранимой процедуры, настроенному на этой странице. При добавлении новых блоков процедур, ссылающихся на одну и ту же хранимую процедуру, имена блоков в дереве индексируются.

В таблице *Параметры процедуры* задаются параметры процедуры, их имена и типы. Количество и порядок следования параметров в таблице *Параметры процедуры* и параметров в хранимой процедуре должны совпадать. Для строковых параметров необходимо указывать максимальную длину значения этого параметра. Параметр процедуры может быть входом, выходом или и входом/выходом (задается в столбцах *Вход* и *Выход*).

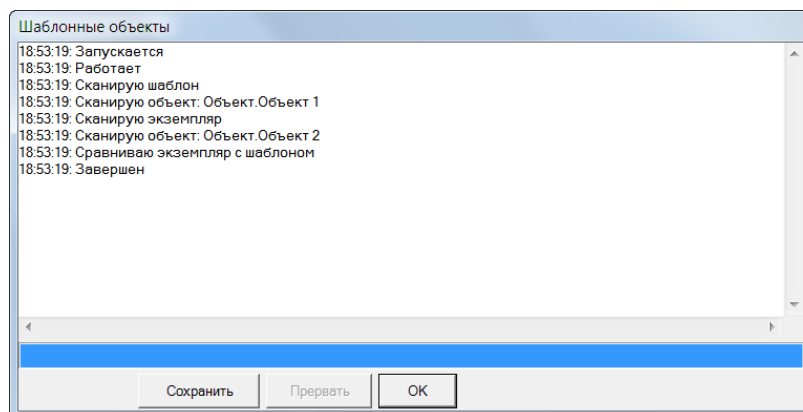
Флаг *Результат* устанавливает возвращает ли процедура какое-либо значение (тип которого также можно установить). При установке флага *Использовать архивные значения* изме-

няется алгоритм выполнения блока процедуры. Если параметр процедуры связан с архивируемой переменной, то в процедуру передаются не текущие значения этого параметра, а архивные. При очередном выполнении процедуры определяется ближайшее время архивного значения, которое еще не было передано ранее (по всем архивным параметрам). Вызывается процедура с архивными значениями по всем параметрам за это время (+-Временная мертвая зона). Это время передается в процедуру как параметр с индексом *Временной параметр*. Временной параметр доступен даже, если нет флага *Выборка*.

В таблице *Выборка* можно задать поля выборки и их тип. Порядок следования параметров в этой таблице и полей в выборке, которую возвращает хранимая процедура, должны совпадать. Имена параметров выборки не обязаны совпадать с именами полей выборки в хранимой процедуре. Привязка осуществляется по позициям (порядковым номерам) для повышения эффективности. Если стоят флаги *Использовать выборку* и *Временной параметр*, то значения, возвращенные через выборку, записываются с меткой времени, передаваемой в этом столбце выборки. Если стоит флаг *Параметр качества*, то значения, возвращенные через выборку, записываются с признаком качества, которые передается в этом столбце выборки. При установке флага *Управление выполнением* в дереве системы у блока процедур появляется вход *Выполнять*. Выполнение процедуры производится только, если на этом входе значение *Истина*.

Страницы свойств Объектов

Закладка *Экземпляры* отображает дерево различий между объектом-экземпляром и его объектом-шаблоном. При переходе на страницу свойств *Экземпляры*, MasterSCADA сканирует проект, чтобы выявить различия между объектом-шаблоном и экземпляром. За ходом проверки можно наблюдать в диалоговом окне *Шаблонные объекты*.



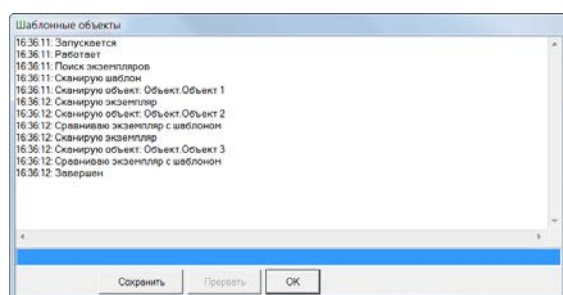
После нажатия кнопки *Ок* страница свойств приобретает вид:



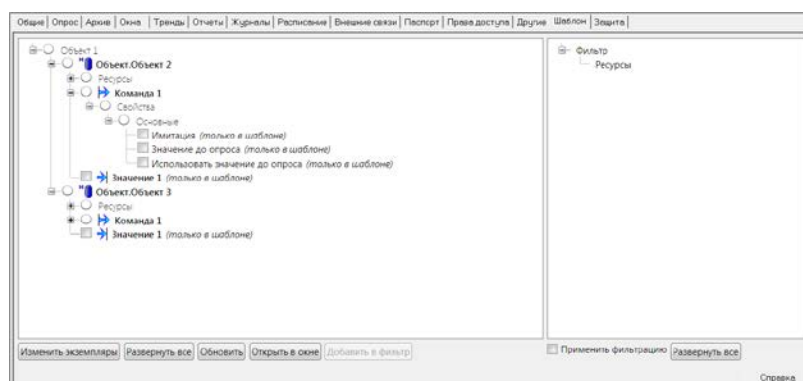
В левой части страницы свойств отображается дерево различий между экземпляром и его шаблоном, напротив каждого элемента дерева можно установить флаг. Элементы отмеченные флагами будут синхронизированы. В правой части страницы находится дерево фильтра исключений. Перетаскиванием мыши или специальной кнопкой можно добавить элементы не подлежащие синхронизации – тогда они будут скрыты в дереве отличий. Настройки фильтра для каждого объекта сохраняются. Использование фильтрации позволяет ускорить внесение изменений в требуемые элементы дерева объектов. Элементы управления представлены в таблице ниже:

Элемент	Описание
Кнопка Изменить экземпляр	Вносит изменения, отмеченные флагом, в данный экземпляр.
Кнопка Изменить шаблон	Вносит изменения, отмеченные флагом, в шаблон из экземпляра. Другие экземпляры данного шаблона автоматически не изменятся после этого действия.
Кнопка Развернуть все	Раскрывает дерево изменений.
Кнопка Обновить	Заново запускает процесс поиска различий между данным экземпляром и его шаблоном.
Кнопка Открыть в окне	Открывает содержимое закладки в новом окне.
Кнопка Добавить в фильтр	Добавляет в фильтр выделенный элемент. Добавлять элементы в фильтр также можно использовать перетаскивание мышью.
Поле Применить фильтрацию	Применяет фильтр к дереву изменений.

Закладка **Шаблоны** отображает дерево различий между объектами-шаблонами и объектами-экземплярами. При переходе на страницу свойств **Шаблоны**, MasterSCADA сканирует проект, чтобы выявить все связанные с данным шаблоном экземпляры и найти различия между ними. За ходом проверки можно наблюдать в диалоговом окне Шаблонные объекты.



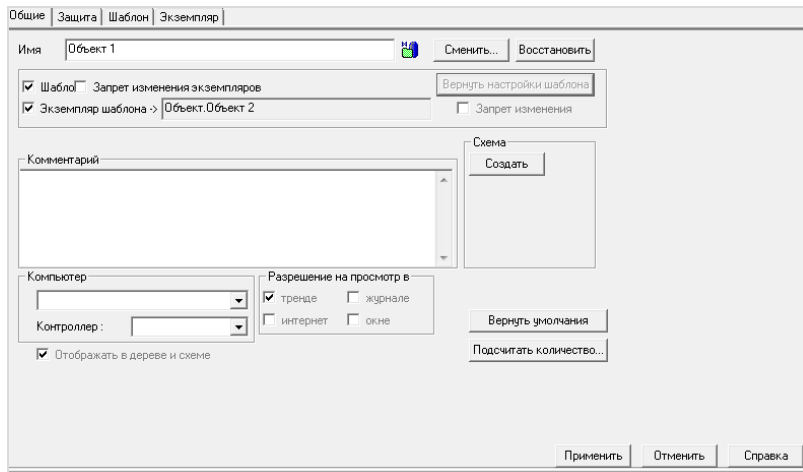
После нажатия кнопки **Ок** страница свойств приобретает вид:



В левой части страницы свойств отображается дерево различий между шаблоном и его экземплярами, напротив каждого элемента дерева можно установить флаг. *Элементы* отмеченные флагами будут синхронизированы. В правой части страницы находится дерево фильтра исключений. Перетаскиванием мыши или специальной кнопкой можно добавить элементы не подлежащие синхронизации – тогда они будут скрыты в дереве отличий. Настройки фильтра для каждого объекта сохраняются. Использование фильтрации позволяет ускорить внесение изменений в требуемые элементы дерева объектов. Элементы управления представлены в таблице ниже:

Элемент	Описание
Кнопка Изменить экземпляры	Вносит изменения, отмеченные флагом во все экземпляры данного шаблона.
Кнопка Развернуть все	Раскрывает дерево изменений.
Кнопка Обновить	Заново запускает процесс поиска экземпляров данного шаблона и различий между ними.
Кнопка Открыть в окне	Открывает содержимое закладки в новом окне.
Кнопка Добавить в фильтр	Добавляет в фильтр выделенный элемент. Добавлять элементы в фильтр также можно используя перетаскивание мышью.
Поле Применить фильтрацию	Применяет фильтр к дереву изменений.

Закладка свойств объекта **Общие** предоставлена на картинке ниже.



Элемент	Описание
Поле Имя	В этом поле задается имя объекта.
Сменить	Изменяет значок объекта (размер файла не должен превышать 32 Кб).
Восстановить	Устанавливает значок объекта по умолчанию.
Поле Комментарий	Необходимо для задания текста комментария.
Список Компьютер	Позволяет назначить объекту компьютер, на котором будет исполняться схема функциональных блоков данного объекта. При выборе <i>На всех компьютерах</i> , объект параллельно будет исполняться на всех компьютерах текущего проекта.
Список Контроллер	Позволяет выбрать контроллер, на котором будет выполняться схема функциональных блоков данного объекта. Список доступных контроллеров определяется выбранным компьютером. На одном контроллере может выполняться несколько схем фб, причем порядок их вычисления в контроллере соответствует порядку следования соответствующих объектов в дереве.
Флаг Отображать в дереве и схеме	Флаг задает необходимость отображения в дереве функциональных блоков и в схеме, а также переменных проекта, входящих в данный объект при использовании Интернет-клиента, а также при вызове диалогового окна вставки переменных на тренд в режиме исполнения.
Флаг Разрешение просмотр в...(тренде, окне, журнале, интернете)	Позволяет разделить область видимости объекта для различных мест. Например, в тренде данный объект должен быть виден, а в интернет - клиенте, в журнале, в окне - нет.
Кнопка Вернуть умолчания	При нажатии на эту кнопку все настройки задаваемые на странице наследуются у объекта родителя. Если объект является корневым (не имеет родителя), то настройки наследуются у компьютера, который назначен для этого объекта. Если объекту не назначен компьютер, то наследование осуществляется из свойств системы.
Кнопка Открыть	Открывает редактор схем функциональных блоков для текущего объекта. В случае если схемы еще не существует, то она создается.
Кнопка Удалить	Удаляет схему текущего объекта.
Кнопка Настроить	Открывает диалоговое окно настройки схемы.
Кнопка Вернуть умолчания	Устанавливает настройки схемы в соответствии с системными настройками.
Флаг Шаблон	Делает объект, которому принадлежит страница свойств шаблоном. объект шаблон является эталонным объектом одного или более объектов экземпляров. После установки данного флага у объекта появляется страница свойств шаблоны.
Флаг Экземпляр шаблона	Экземпляр шаблона показывает к какому шаблону относится данный экземпляр. Можно переназначить данное поле, перетаскив в него объект, назначенный шаблоном. У объекта, являющегося шаблоном, есть настройка запрет изменения экземпляров, которая позволяет защитить экземпляры от изменения, При включенном флаге изменения можно вносить только в шаблон. У экземпляров есть настройка запрет изменения, которая позволяет переопределить возможность внесения изменений для конкретного экземпляра.
Кнопка Вернуть настройки шаблона	Возвращают настройки шаблона закладки общие.
Кнопка Подсчитать количество элементов...	Открывает окно, в котором указано, какие элементы присутствуют в данном объекте.

На закладке **Опрос** устанавливаются параметры опроса.

Элемент	Описание
Периодический	Если у объекта нет собственного цикла вычислений, он пересчитывается в цикле объекта-родителя. Если нет необходимости пересчитывать объект каждый родительский цикл (такт), то можно установить кратность в окне каждый ____ такт.
По изменению входов	Как только данные на входах обновились объекту присваивается особое свойство как нуждающемуся в пересчете и как только подойдет его очередь, он немедленно будет пересчитан. С некоторой погрешностью можно считать, что это происходит сразу после обновления данных на входах. Верхнее поле мертвая зона - в процентах, нижнее поле мертвая зона - в физических единицах.
Вернуть умолчания (кнопка)	Установить параметры объекта-родителя. Для главного объекта - собственный цикл с периодом назначенного компьютера или, если не назначен компьютер, системы (с периодом передачи данных на одном компьютере).
Флаг Восстановление при рестарте	Используется только для наследования подчиненными элементами.
Флаг Собственный цикл	Устанавливает собственный цикл вычислений. Поле период - период собственного цикла вычислений. Список приоритет - если задавать значения > 100, то эти задачи будут получать больше процессорного времени в случае его нехватки. Также более точно будет выдерживаться период цикла.
Поле Условие выполнения	Задаёт условие, по которому будет выполняться опрос. При значении <i>Истина</i> опрос выполняется как обычно, при значении <i>Ложь</i> опрос не выполняется. В данном поле используются только дискретные переменные.
Поле По переднему фронту	Дополнительное безусловное выполнение по импульсу. В данном поле используются только дискретные переменные.
При значении <i>Ложь</i> у переменной поля <i>Условное выполнение</i> стандартный опрос отключен, но возможен единовременный опрос при подаче импульса в переменную поля <i>По переднему фронту</i> .	

Лист **Восстановление при рестарте** настраивается аналогично свойствам системы.

На закладке **Типизация** объекта производится назначение списка экземпляров типовому объекту.

Элемент	Описание
Поле Список экземпляров	Список с выбором доступных экземпляров.
Флаг Циклическое переключение экземпляров	Если настройка включена, то переключение опроса экземпляров происходит автоматически, то есть происходит циклический опрос всех экземпляров списка. Период цикла выдерживается для каждого экземпляра в отдельности.
Поле Условие переключения	Если условие <i>Истина</i> то выполняется циклический опрос экземпляров объекта.
Поле Команда пе-	Команда строкового типа, в которой задается имя экземпляра, на который необходи-

Элемент	Описание
реключения экземпляров	мо переключиться. Может работать совместно с циклическим опросом - при задании команды можно переключить на один такт на экземпляр команды, а затем продолжить циклическое переключение, начиная с экземпляра команды, то есть можно использовать для внеочередного принудительного опроса.
Поле Текущий экземпляр	В данное поле можно перетащить переменную (команду, значение, вход ФБ) - в эту переменную будет записываться имя текущего (то есть опрашиваемого) экземпляра.
Поле Максимальное количество экземпляров	Показывает количество экземпляров, которое будет выполняться для данного объекта.

В отличие от закладки **Окна** у свойств **Системы**, где создаются типы документов, здесь можно создавать и удалять сами документы для **Объекта**.

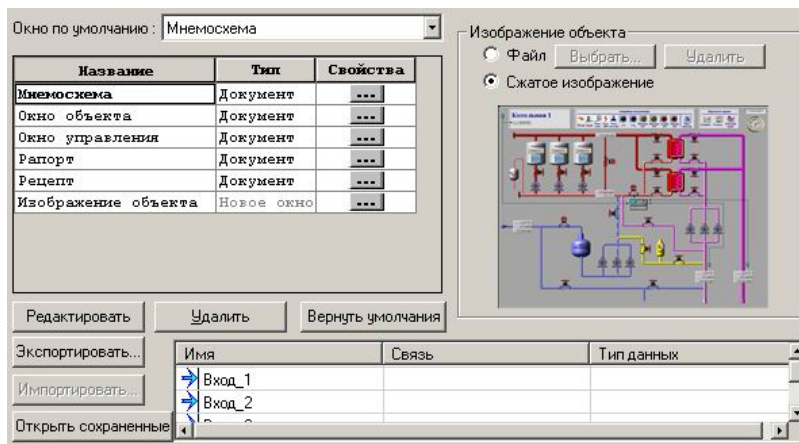


Таблица содержит перечень мнемосхем, доступных для редактирования. Если существует документ определенного типа, соответствующая строчка отображается жирным шрифтом. Чтобы выделить окно, щелкните по нему мышью, и оно выделится в рамку. Для каждой мнемосхемы можно переопределить способ открытия, а также расположение на экране. Настройки по умолчанию задаются на странице свойств системы **Окна**, или компьютера (если объекту назначен компьютер):

- Столбец *Название* - имя типа мнемосхемы;
- Столбец *Тип* - с помощью выпадающего списка для каждого окна можно настроить способ его открытия;
- Столбец *Свойства* - при нажатии на кнопки, находящиеся в ячейках этого столбца открывается диалоговое окно, с помощью которого для каждой мнемосхемы можно настроить параметры окна (расположение на экране и т.п.).

Выпадающий список *Окно по умолчанию* позволяет выбрать окно, которое будет устанавливаться по умолчанию при создании кнопки перехода к документам данного объекта. При нажатии кнопки *Редактировать* открывается окно для редактирования данного документа.

Кнопка *Открыть сохраненные* позволяет просмотреть содержимое папки, в которой хранятся графические файлы с изображениями мнемосхемы, созданные в режиме исполнения. Графическое *Изображение* на кнопке объекта в мнемосхеме другого объекта. Это может быть сжатое изображение мнемосхемы данного объекта или графический файл (задается с помощью переключателей *Файл* и *Сжатое изображение*). Для выбора файла нажмите соответствующую кнопку и выберите любое изображение в формате BMP. Объект может быть использован в мнемосхеме для того, чтобы можно было быстро перейти к его документам. Остальные элементы управления производят аналогичные описанным ранее действия.

Закладка **Тренды** - отображение графиков изменения данных технологического процесса с течением времени. В MasterSCADA совмещен просмотр архивных (исторический тренд) и текущих (тренд реального времени) данных на одном графике. С помощью кнопок можно добавлять, изменять и удалять тренды для объекта. Для того чтобы значения переменной объекта появились в тренде достаточно перетащить ее из дерева объектов в окно тренда.

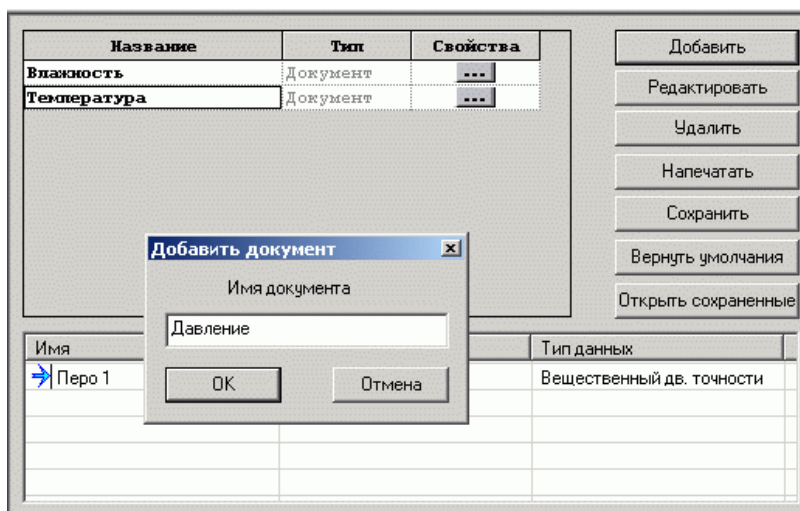


Таблица содержит перечень созданных трендов, доступных для редактирования. Для каждого тренда можно переопределить способ его открытия, а также расположение на экране. Настройки по умолчанию задаются на странице свойств системы окна, или компьютера (если объекту назначен компьютер).

Для создания нового тренда необходимо нажать кнопку *Добавить*. Сначала появится окно для задания имени новому тренду, затем откроется сама подсистема трендов для редактирования нового документа. Кнопка *Редактировать* открывает уже существующий тренд. Кнопка *Открыть сохраненные* позволяет просмотреть содержимое папки, в которой хранятся файлы с изображениями трендов, созданные в режиме исполнения. Кнопка *Сохранить* - сохранить тренд в виде файла в формате JPG. Остальные элементы управления выполняют функции аналогичные описанным ранее.

Для создания **Отчетов** в MasterSCADA (закладка отчёты) используется либо генератор отчетов, либо Excel, удобное средство редактирования таблиц. Для того чтобы использовать в отчете (таблице Excel) текущие значения переменных достаточно перетащить мышью вход или выход в нужную ячейку таблицы, в режиме исполнения значения будут записываться в эту ячейку автоматически. При перетаскивании переменной по правой кнопке мыши, в таблице будет отображено имя переменной просто как текст, также есть возможность перетаскивания любых других элементов деревьев проекта.

Для создания часовых, суточных, сменных отчетов используется *ФБ Данные для рапорта*. Список содержит перечень имен уже существующих отчетов. Для действий (редактирования, удаления, просмотра) с нужным отчетом выделите его и нажмите кнопку действия справа от списка. В поле *Связи документа* для выделенного отчета отображаются его переменные, а также их связи и типы данных (отсутствует для отчетов Excel). В этом поле производится настройка параметров открытия выделенного отчета.

Описание элементов управления группы *Прочее*:

- Флаг *Закрывать при показе проекта* - документ, открытый с этой настройкой закрывается при нажатии *Показать проект*;
- Флаг *Закрывать другие окна* - когда установлен *Истина*, при открытии текущего документа закрываются все ранее открытые окна;
- Флаг *Не закрывать при открытии нового окна* - когда установлен *Истина*, при открытии текущего окна не закрываются все ранее открытые окна;
- Флаг *Открывать на другом мониторе* - если используется компьютер с двумя видеовыходами, куда можно подключить два монитора, то флаг устанавливается, при необходимости отображения его на другом экране;
- Подгруппа *Размер* - позволяет задать размеры окна отчета. Размер по умолчанию задается на странице свойств *Системы Окна* в строке *Отчет*;
- *Тип* - позволяет настроить способ открытия выделенного отчета.

Группа *Расположение окна* позволяет настроить первоначальное расположение выделенного отчета при открытии:

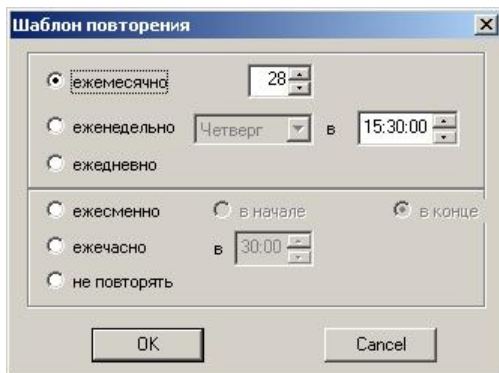
- *Стандартное* - при открытии окно располагается в левом верхнем углу экрана;
- *По центру* - окно располагается по центру экрана;

- **Смещение** - позволяет для открываемого окна задать смещение в пикселях относительно левого верхнего угла экрана. Смещение становится доступна, когда для выделенного отчета в графе *Расположение окна* установлено *Смещение*.

В поле *Заголовок* - вводится название окна. Флаг *Отображать заголовок* - при открытии выбранного отчета отображается заголовок. Флаг *Отображать рамку* - если флаг установлен, то в режиме исполнения отображается рамка объекта. Флаг *Поверх других окон* - при открытии окно располагается поверх других окон. Флаг *Разрешить закрытие* - если флаг *Ложь*, то окно закрывается только в случае остановки проекта. Флаг *Разрешить изменение размера* - позволяет в режиме исполнения изменять размер окна.

- ❖ **Важно!** Для корректного отображения значения переменных типа *Время*, необходимо вручную настроить ячейку в *Excel*, которая будет принимать данные из *MasterSCADA*. Сделать это можно как до, так и после операции перетаскивания переменной, выбрав нужный формат отображения данных (*Время* или *Дата*) для целевой ячейки.

Закладка **Расписание** - удобное средство для управления действиями во время работы программы, например, автоматически еженедельно в установленное время печатать рапорт. Выберете в списке, (всплывающем при нажатии на соответствующую ячейку таблицы) действия то, что необходимо сделать с документом. Затем выберите (также из списка) документ, с которым необходимо выполнять это действие. Затем выберете, какое действие должно быть выполнено. При нажатии на соответствующее поле колонки *Время*, появится календарь и часы, где можно установить нужные значения. И, наконец, щелкните на ячейке в колонке *Повтор* и с помощью шаблона установите нужные параметры повторов действия:



С помощью этого диалогового окна можно установить необходимую периодичность выполнения некоторого действия:

Действие	Документ	Параметр	Время	Повтор
Открыть	Инемосхема		В 16:42	ежедневно
Присвоить	Объект 1. Значение: 100			ежечасно

- **Ежемесячно** - действие будет повторяться ежемесячно, в указанный день и время. Если в текущем месяце указанного дня нет, то действие будет выполняться в последний день месяца;
- **Еженедельно** - для задания периодичности выполнения действия задается день недели и время;
- **Ежедневно** - действие выполняется каждый день в указанное время;
- **Ежесменно** - действие выполняется либо в начале, либо в конце каждой смены, определенной для данного рабочего места;
- **Ежечасно** - действие выполняется каждый час в указанное время.

Для действия над документами в левой колонке таблицы нужно выбрать из всплывающего списка одно из действий. Затем во второй колонке документ, причем можно выбрать только один из документов, принадлежащих данному объекту. Для действия над переменной, выберите действие *Присвоить*, затем перетащите мышью переменную в таблицу из дерева объектов, ее имя отобразится во второй колонке, затем в третьей колонке введите присваиваемое значение, либо перетащите туда переменную - источник. Также на выходе некоторой переменной можно выдать импульс. Для этого в списке действий выберите *Выдать импульс*, и затем перетащить мышью нужную переменную из дерева объектов во вторую колонку таблицы. Для *Синхронизации времени* в поле действие нужно выбрать *Выполнить*, после чего в поле *Документ* появится синхронизировать время. В поле параметр вводится имя сервера по которому синхронизируется время.

На закладке **Паспорт** на основе шаблонов создаются экземпляры **Паспортов**.

Группа	Параметр	Тип	Значение	Запис	Сравнение	Уставка	Запис	Сообще
Датчики	P	Аналоговый		<input checked="" type="checkbox"/>	>		<input checked="" type="checkbox"/>	Работа
Насосы	Q	Аналоговый		<input checked="" type="checkbox"/>	=		<input checked="" type="checkbox"/>	

Описание элементов управления:

- Выпадающий список **Шаблон** - содержит список шаблонов из той категории, к которой относится редактируемый элемент проекта (для которого открыта данная страница свойств);
- Таблица параметров шаблона - содержит список параметров выбранного шаблона. Для редактирования доступны все поля, за исключением полей **Группа**, **Параметр** и **Тип**, которые определяются в шаблоне. Значения полей, унаследованные из шаблона, отображаются в таблице серым цветом, переопределенные - черным. **Значение** и **Уставка** могут быть переменными, для установки связи необходимо перетащить необходимый элемент из дерева объектов или дерева системы в соответствующее поле;
- Кнопка **Импорт** - импорт возможен только при наличии выбранного шаблона и производится по следующей схеме: для каждого параметра шаблона производится поиск среди параметров импортируемого файла. Если найден параметр с таким же именем (учитывается **Имя группы** и **Имя параметра**), то поля параметра шаблона заполняются на основе полей импортируемого параметра;
- Кнопка **Установить умолчания** - восстанавливает установки шаблона;
- Группа **Имя окна** - служит для определения заголовка окна паспорта, при выборе радиокнопки **Другой текст** появляется поле ввода, где пользователь может написать свой текст заголовка.

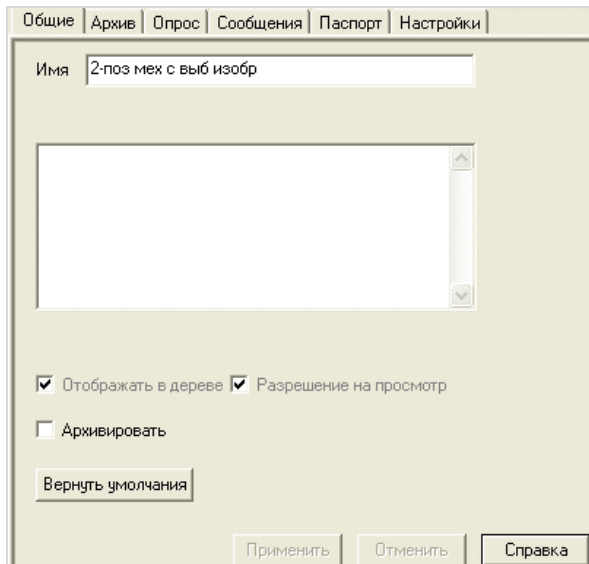
На закладке **Другие** осуществляется подключение к объекту внешних документов.

Название
Диаграмма Microsoft Excel
Документ WordPad
Точечный рисунок

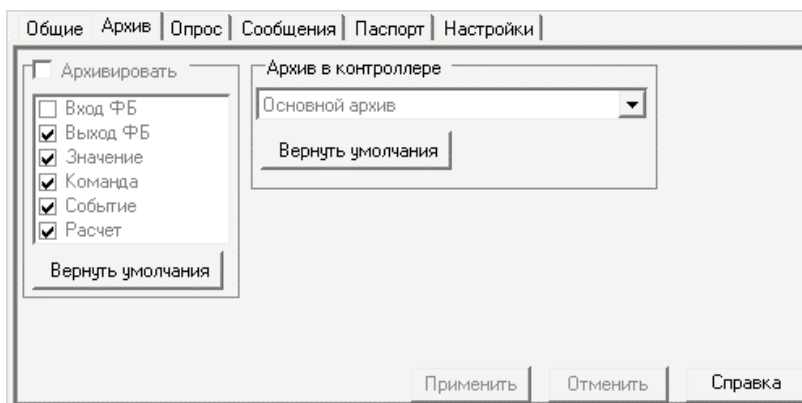
Для подключения к проекту документа, нажмите **Добавить**. Затем выберите из списка, который будет предложен, нужный тип документа. Для создания и изменения этого списка используется **Организер документов**. Документ будет автоматически создан и открыт для редактирования. После внесения изменений сохраните документ (меню **Документ** - сохранить или Ctrl+S), а затем закройте его чтобы перейти обратно в проект. Также вернуться в проект можно через команду **Показать** из меню **Проект**. Чтобы изменить содержание документа впоследствии, можно воспользоваться кнопкой **Редактировать**. Кнопка **Редактировать** открывает соответствующую программу для редактирования документа. можно работать с документом, например с таблицей Microsoft Excel, не покидая MasterSCADA. Нажатие кнопки **Удалить** приведет к удалению документа из проекта.

Страницы свойств Функциональных блоков

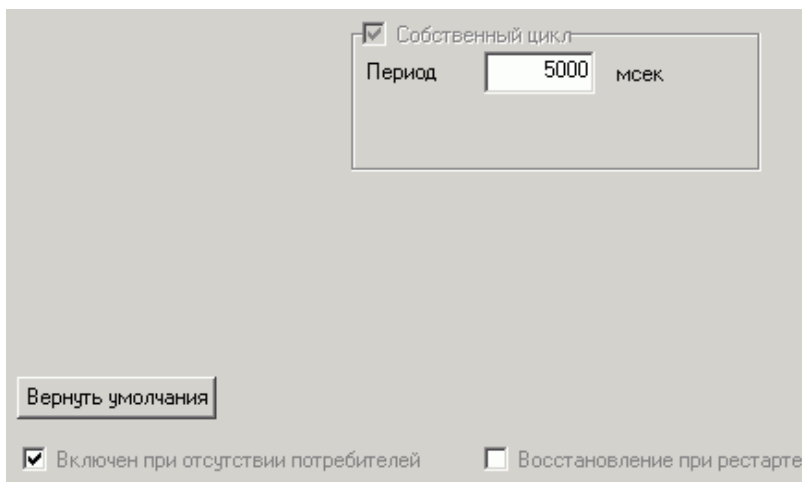
На страницах свойств возможно настраивать или получать информацию о некоторых параметрах функциональных блоков. Конкретные возможности настроек имеются в описании соответствующих закладок страниц свойств. Кроме того, функциональные блоки могут иметь индивидуальные страницы свойств, запрограммированные разработчиком функционального блока. Все не описанные элементы управления в данном разделе аналогичны элементам управления описанным ранее.



Закладка **Архив** на рисунке ниже. Выпадающий список *Архив в контроллере* - выбирается архив, где будут храниться параметры ФБ. Данное поле появляется на закладке *Архив* в том случае, если параметры ФБ архивируются в контроллере и в настройках контроллера для параметра *Количество дополнительных задач архивирования* установлена одна или более дополнительных задач.



На закладке **Опрос** настраиваются свойства опроса функционального блока. Если нажата кнопка *Периодический*, то пересчет Функционального блока будет производиться в цикле родительского объекта, (объекта, содержащего этот ФБ) с периодом опроса этого объекта. Если *По изменению входов* - при обновлении данных хотя бы одного входа, функциональному блоку будет присвоен признак того, что он требует пересчета. С небольшой погрешностью можно считать, что пересчет произойдет сразу после изменения входов.



Вид закладки *Настройки* изображён на рисунке ниже.

Группа *Выход Авария формировать*:

- *От входа Авария* - сигнал авария формируется от входа авария, т.е. вне функционального блока;
- *По несрабатыванию за ___ сек* - сигнал авария формируется внутри ФБ по несрабатыванию за максимальное время срабатывания.

Группа *Режим при старте*:

- *Дистанционный* - устанавливает дистанционный режим работы при старте;
- *Автоматический* - устанавливает автоматический режим работы при старте.

Страницы свойств *Переменных-входов*

К переменным типа *Вход* относятся:

- Значения;
- Входы функциональных блоков;
- OPC - переменные, имеющие разрешение на чтение.

В зависимости от типа они могут иметь разный набор закладок. Вид закладки **Общие** представлен на рисунке ниже.

В группе *Тип* указывается тип данных. Для OPC переменных для записи и входов функциональных блоков тип данных изменить невозможно. Таким образом, возможность изменения типа данных существует только у *Значений* (если у него установлен флаг *Запрет наследования типа*). Флаг *Запрет наследования типа* доступен только у *Значений*. По умолчанию, значения наследуют тип данных переменной-источника. При помощи этого флага, можно установить для значения тип данных, отличный от типа источника. Все переменные входы, за исключением значения (если у него установлен флаг запрет наследования типа) наследуют диапазон изменения у переменной-источника. Для значения из этого списка можно выбрать диапазон изменения значения. Диапазоны настраиваются на странице свойств системы *Шкалы*.

На закладке **Опрос** можно выбрать один из трех видов формирования значений входа:

- *Опрос* - только если вход имеет связь;
- *Константа* - постоянное значение на входе;
- *Имитация* - значения на входе изменяются по выбранному закону.

Поле *Присоединённый выход* содержит полное имя выхода, с которым связан данный вход (если связь существует). Рядом указывается тип входа, который устанавливается согласно типу присоединенного выхода (для функциональных блоков и OPC переменных производится преобразование типов).

Кнопка *Сброс* разрывает связь с выходом. Нажатие кнопки *По изменению* позволяет задавать параметры мёртвой зоны. При установленном флаге *Значение до опроса* после перехода в режим исполнения переменная имеет заданное в поле ввода значение вплоть до момента получения значения от присоединенного выхода.

В поле *Константа* устанавливается постоянное значение на входе (или выбирается из раскрывающегося списка в случае, если тип - логический). Флаг *Имитация* - на вход подаются значения, изменяющиеся по одному из законов имитации. Параметры законов имитации - периоды и пограничные значения, устанавливаемые по умолчанию, настраиваются в свойствах системы - закладка *Опрос*, лист *Имитация*

- *Закон* (раскрывающийся список) - выбирается закон имитации (пила, синус и т. д.);
- *Минимум* - минимальное значение изменяющейся величины;
- *Максимум* - максимальное значение.

В режиме разработки закладка *Проверка* выглядит следующим образом:

- ❖ **Важно!** OPC-переменные могут иметь определенные свойства, если это предусмотрено разработчиком. Какие это свойства - зависит от конкретного OPC сервера и OPC переменной.

Закладка *Архив*.

Элемент	Описание
Группа Экспорт в БД	БД-коннектор - БД-коннектор, использующийся для хранения архивов по данной переменной. Имя в БД - имя параметра в базе данных (по умолчанию - путь к переменной). Вернуть умолчания - устанавливает стандартное имя в базе данных.
Группа Параметры архивирования	Параметры наследуются от выбора шаблона на закладке общие.
Группа Переопределить настройки шаблона	Периодически - архивировать с заданным периодом (если = 0, то пишутся значения на каждом опросе, при выходе из режима исполнения не будет записываться последнее полученное значение в архив). По изменению - архивировать при изменении на мертвую зону.

В зависимости от режима работы и типа переменной некоторые элементы закладки **ДАННЫЕ**, в которых нет необходимости, могут отсутствовать.

Среднее время	Время	Качество	Значение	Данные за (минут)
1004 мс	16:47:01.701 12/04/06	Норма	13.958	10
	16:47:00.701 12/04/06	Норма	12.292	
	16:46:59.701 12/04/06	Норма	10.625	
Минимальное время	16:46:58.686 12/04/06	Норма	8.933	
735 мс	16:46:57.686 12/04/06	Норма	7.267	
	16:46:56.686 12/04/06	Норма	5.600	
Максимальное время	16:46:55.701 12/04/06	Норма	3.958	
1343 мс	16:46:54.670 12/04/06	Норма	2.240	
	16:46:53.780 12/04/06	Норма	0.757	

Функции элементов управления указаны в таблице ниже.

Элемент	Описание
Тип данных	Указывается тип данных переменной.
Текущие данные	Значение данных переменной в настоящий момент.
Признак недостоверности	ОРС - признак качества данных переменной (если есть).
Время последнего опроса	Время последнего поступления данных.
Заблокировать	Если нужно установить значение переменной, которое будет сохраняться при последующих опросах независимо от данных, пришедших по связям, результатов вычислений и прочих обстоятельств, способных изменить значение переменной, то нужно установить этот флаг. Значение заблокированной переменной в дереве объекта отображается в угловых скобках.
Установить значение	Если необходимо установить какое-то значение переменной, независимо от опроса, нажмите кнопку и в появившемся диалоговом окне введите значение. Кроме этого, можно устанавливать значения переменных прямо из дерева.
Среднее время опроса	
Минимальное время опроса	В этих полях показывается статистика опроса переменной.
Максимальное время опроса	
Окно с данными	Предназначено для просмотра данных переменной.
Данные за последние ___ минут	Указывается за сколько минут получить данные.
Получить	Обновить окно с данными.
Сохранить	Сохранить содержимое окна с данными в файл. В появившемся при нажатии кнопки окне необходимо выбрать папку и указать имя файла, куда будут сохранены данные.
Печать	Напечатать содержимое окна с данными.
Только выделенные	Печатать или сохранять в файл только выделенные данные. Выделение осуществляется левой кнопкой мыши, при этом допускаются сочетания: Ctrl+ЛКМ - добавить к выделенным ранее. Shift+ЛКМ - добавить все между текущим и выделенным ранее.

Страницы свойств Переменных-выходов

К переменным типа *выход* относятся

- Команда;
- Расчет;
- Событие;
- ОРС-переменные, имеющие разрешение на запись;

- Выходы функциональных блоков.

В зависимости от типа они могут иметь различный набор закладок. Вид закладки **Общие** указан на рисунке ниже.

Группа *Тип* - устанавливается тип данных (используется для *Команды*, для остальных типов выходов предопределен и показан в неизменяемом виде). Если величина выхода изменяется в предопределенном диапазоне, то в раскрывающемся списке *Дискретные значения* можно выбрать этот диапазон. Диапазоны настраиваются сразу для всей системы на закладке *Шкалы*. Таким образом, можно избавиться от настройки параметров контроля значений. Помимо этого, данные с установленным диапазоном отображаются вместе с соответствующей единицей измерения

Вид закладки **Опрос** указан на рисунке ниже. Если выбрана кнопка *Периодический* - данные на выход будут передаваться в цикле родительского объекта (объекта, содержащего этот выход) с периодом опроса этого объекта. Если нет необходимости передавать данные на выход каждый родительский цикл (такт), то можно установить кратность в окне *Каждый ___ такт*. *Период* - период опроса родительского объекта. Если выбрана кнопка *По изменению*, то передача данных зависит от типа источника данных и величины мертвой зоны

Расчет и Событие:

- Если данные в расчет или событие поступают в результате вычислений по формуле, то новые данные на выход передаются в том случае, если любые данные, используемые в формуле изменились на величину, большую чем мертвая зона;
- Если данные в расчет или событие поступают со страницы данные или из дерева объектов через окошко *Ввод значения*, по расписанию объекта или по действию события, то новые данные на выход передаются по их изменению, без учета мертвой зоны.

Кнопка *Восстановить умолчания* – возвращает настройки как у родительского объекта.

Закладка **Опрос Команды** имеет несколько другой вид.

Элементы управления аналогичны описанным выше. Но есть и дополнительные элементы. Если выбрана кнопка **По изменению**, то передача данных зависит от типа источника данных и величины мертвой зоны:

- Если для установления данных на выходе команды используется элемент управления в мнемосхеме, созданный с помощью перетаскивания (*Задатчик*, *Редактируемое значение*), а также если данные поступают со страницы *Данные* или из дерева объектов через окошко *Ввод значения*, по расписанию объекта или по действию события, то новые данные на выход передаются по их изменению, без учета мертвой зоны;
- Если для установления данных на выходе команды используется элемент управления в мнемосхеме с динамизированным свойством этого элемента от выхода мнемосхемы, то новые данные на выход передаются в том случае, если данные на выходе мнемосхемы изменились на величину, большую чем мертвая зона.

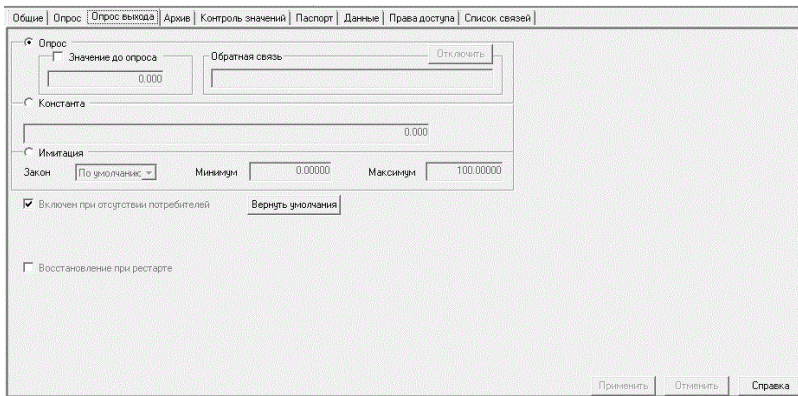
Продолжительность импульсов - кол-во тактов, в течении которых на мнемосхеме кнопка без фиксации будет находиться в нажатом состоянии. *Не передавать данные обратной связи* - данные при настроенной обратной связи не выдаются на выход команды, а только отображаются на мнемосхеме. В этом случае, команда выдает только то значение, которое определил пользователь.

Закладка **Права доступа** позволяет устанавливать индивидуальные права доступа операторов на действие *Установить* значение для *Событий* и *Команд*.

Должность	Лок.	Глоб.	Журнал	Подтверждение
Просмотр	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Оператор	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Старший оператор	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Начальник смены	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Начальник цеха	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Диспетчер	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

В строках таблицы перечислены все имеющиеся в системе должности, а в столбцах напротив соответствующие этим должностям настройки доступа на действие *Установить* значение. По умолчанию, эти настройки для каждой из должностей наследуются из свойств системы (страница свойств операторы). Кнопка *Вернуть умолчания* устанавливает настройки прав доступа по умолчанию (в соответствии с системными настройками, задаваемыми на странице свойств операторы). В поле *Название действия* можно задать имя действия. Оно используется в подтверждении и записи в журнал действий, если оператор нажмет связанную с командой кнопку или изменит значение команды в мнемосхеме.

На закладке **Опрос выхода** выбирается один из трех возможных способов формирования значения переменной.

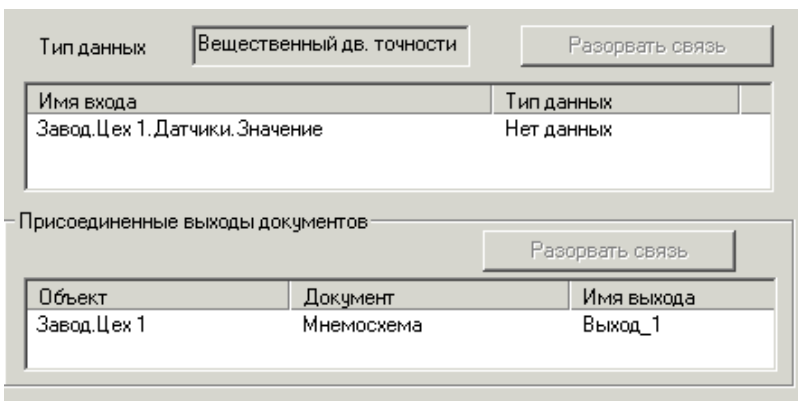


При нажатом флаге *Опрос* значение переменной формируется обычным способом (например функциональным блоком, которому принадлежит выход, если это выход функционального блока). *Значение до опроса* - при установленном флаге после перехода в режим исполнения выход имеет заданное в поле ввода значение вплоть до момента формирования значения выхода в результате вычисления формулы (для расчетов и событий), ручного ввода (для команды), вычисления ФБ (для выходов ФБ).

Обратная связь (только для *Команды*) - в это поле перетаскиваются выход функционального блока или OPC сервера. В режиме исполнения команда с установленной обратной связью отображает последнее изменившееся значение - от оператора или пришедшее по обратной связи. *Константа* - независимо от величины вычисленного значения, выход будет равен заданной величине константы.

Флаг *Приоритетный параметр* - используется в случае программирования контроллеров модулем MasterPLC. Если данный флаг установлен, то переменная опрашивается с периодом, указанным в поле *Период приоритетного опроса* на странице свойств контроллера *Настройки*. Остальные элементы управления аналогичны описанному ранее.

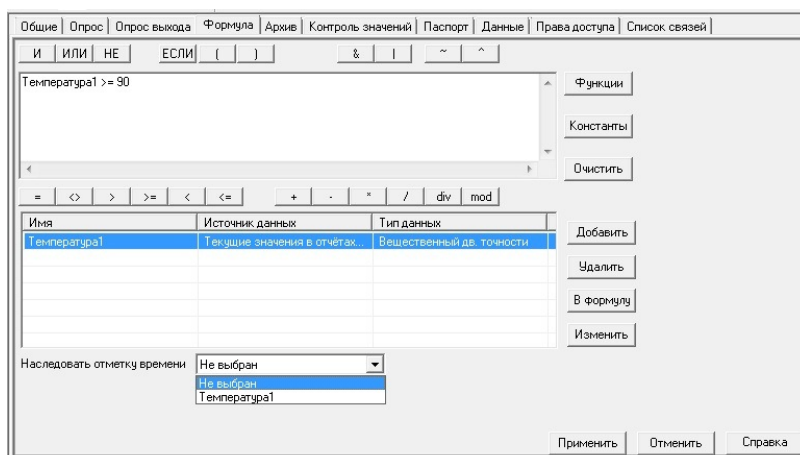
Закладка *Связи* существует только у выходов, имеющих одну или несколько связей. Она содержит информацию обо всех входах, связанных с данным выходом.



Функции элементов управления описаны в таблице ниже.

Элемент	Описание
Поле Тип данных	Тип данных данного выхода.
Таблица связей с переменными	Список входов (значений), с которыми данная переменная имеет связь.
Присоединенные выходы документов	Список выходов документов, с которыми данная переменная имеет связь.
Кнопка Разорвать связь	Чтобы разорвать связь с каким-либо входом или документом нужно выделить его в таблице и нажать эту кнопку.

Закладка *Формула* позволяет сформировать значение выхода произвольным образом, удобным пользователю. Для *Событий* тип результата вычислений должен быть логическим.



Верхнее окно предназначено для ввода и редактирования формулы. Вокруг него расположены кнопки для ввода функций и операторов. Нижнее окно - таблица переменных, это - список переменных, которые можно использовать в формуле. Для редактирования формулы предлагается набор арифметических, логических, побитовых операций и операций сравнения, а так же набор функций и констант. Редактирование формулы производится в верхнем окне и возможно любым способом: набором с клавиатуры или использованием соответствующих кнопок. Синтаксис формулы и схож с любым языком программирования, в случае его нарушения выводится соответствующее сообщение. Порядок действий - как в обычной арифметике.

Допускается использовать:

- Числа (в качестве десятичной запятой используется точка, на дополнительной клавиатуре десятичная точка на русском языке не будет работать, так как пишется запятая, переключите язык или используйте основную клавиатуру);
- Константы;
- Имеющиеся функции и операции;
- Имена переменных, описанные в таблице (можно русские).

Список типов переменных, используемых в формуле: Целый, Вещественный, Логический, Время. Автоматически выполняется приведение значения целого типа к значению вещественного типа. Для обратного преобразования необходимо использовать функции `floor` и `ceil`. Вставка переменной в формулу осуществляется в два этапа:

- С помощью кнопки *Добавить* завести новую переменную;
- Выделить имя переменной и нажать кнопку *В формулу* (можно набрать имя переменной в окне с клавиатуры).

С помощью кнопки *Удалить* переменные удаляются из списка. Убедитесь, что удаляемая переменная не используется в тексте формулы, иначе возникнет синтаксическая ошибка. Нажатие кнопки *Изменить* выводит диалог для настройки переменной. Для того чтобы использовать в качестве источника данных для переменной, используемой в формуле вход или выход ФБ или переменную объекта нужно навести мышью в дереве объектов на переменную, нажать левую клавишу и не отпуская ее перетащить выход в таблицу переменных. Если перетащите на строчку с уже имеющейся свободной переменной, то автоматически установится связь. Если на свободное место или на занятую переменную, то будет создана новая переменная.

Изменение типа источника данных в дереве автоматически не приводит к изменению типа переменной в формуле. Для наследования типа источника нужно выделить переменную в списке, нажать кнопку *Изменить*, и в выпадающем списке выбрать пункт *Унаследовать тип*. Если тип переменной совпадает с типом источника данных, то название типа в таблице отображено серым цветом, если не совпадает - черным. Переменные *Расчет* и *Событие* могут наследовать метку времени одной из переменных включенных в формулу. Для этого необходимо выбрать в поле *Наследовать отметку времени* необходимый параметр.

При переходе выхода *События* или дискретной *Команды* из *Ложь* в *Истина* может выполняться действие. На закладке *Действие* настраивается какое или какие действия выполнить.



Для добавления нового действия нужно нажать кнопку *Добавить*. Для удаления уже существующего действия нужно его выделить и нажать кнопку *Удалить*. При установленном флаге *Использовать документы компьютера* все действия, связанные с документами, будут производиться с документами *Компьютера*, а не *Объекта*.

Для *действия над документами* - в левой колонке таблицы выберите из всплывающего списка одно из действий. Затем выберите во второй колонке документ, причем можно выбрать только один из документов, принадлежащих данному объекту.

Для *присвоения определенного значения переменной*, либо значения другой переменной, выберите действие *Присвоить*, затем перетащите мышью переменную в таблицу из дерева объектов, ее имя отобразится во второй колонке, затем в третьей колонке введите присваиваемое значение, либо перетащите туда переменную-источник. Также на выходе некоторой переменной можно выдать импульс. Для этого в списке действий выберите *Выдать импульс*, и затем перетащите мышью нужную переменную из дерева объектов во вторую колонку таблицы.

Кроме действий, определенных в MasterSCADA, также могут выполняться внешние команды, под которыми понимается запуск исполняемых модулей с диска. Для настройки внешней команды, в списке действий *События* или *дискретной Команды* выберите *Выполнить*. При этом, из выпадающего списка в столбце *Объект* нужно выбрать имя исполняемого файла, а в столбце *Параметр* можно указать дополнительные параметры, необходимые для его запуска. Все модули, которые планируется запускать по действию события или дискретной команды должны находиться в папке *Commands*, которая находится в системной папке MasterSCADA (путь по умолчанию - *C:\MasterSCADA Projects*). В комплект поставки MasterSCADA входит единственная внешняя команда - *Синхронизировать время*, с помощью которой производится синхронизация времени между компьютерами в сети. Имя сервера, по которому будет синхронизировано время задается с помощью параметра.

Чтобы при наступлении (переходе результата из *Ложь* в *Истина*) события выводилось какое-либо *Сообщение*, его необходимо создать. Создание сообщений происходит на закладке **Сообщения** указанной на рисунке ниже.

Шаблон сообщения:

Какой-то текст {Переменная1} Еще какой-то текст {Переменная2}

В текст

Имя	Источник данных	Тип данных
Переменная1	Объект.Участок 3.Расчет	Вещественный дв. точности
Переменная2	Объект.Участок 3.Команда	Вещественный дв. точности

Добавить Удалить

Категория:

Приоритет:

Необходимость квитирования

Вернуть умолчания

Жакал

- Сообщение
- Строка статуса
- Журнал
- Принтер
- Архив

В окне *Шаблон сообщения* набирается текст сообщения. Если в тексте сообщения должны присутствовать значения переменных (кроме входов ФБ, т к они являются источниками данных только для своего ФБ), нужно перетащить мышью переменную из дерева объектов в таблицу переменных или поставить курсор в то место в тексте, где должно быть значение, выделить нужную переменную в таблице и нажать кнопку *В текст*. Затем установить *Категорию*, *Приоритет*, *Необходимость квитирования* и выбрать *Канал вывода* для сообщения.

Закладка **Контроль значений** существует только у аналоговых *Расчетов*, *Команд*, *выходов ФБ* и *ОРС переменных для чтения*. Если значение выхода выходит за пределы указанных границ, ему присваивается соответствующий ОРС признак качества и формируется соответствующее системное сообщение.

The screenshot shows a configuration window with the following fields and values:

- Верхняя аварийная граница: 1000
- Верхняя предупредительная граница: 990
- Нижняя предупредительная граница: 10
- Нижняя аварийная граница: 0
- "Мертвая зона": 1.5
- Максимальная скорость изменения: 2.5
- Button: Вернуть умолчания

В полях *группы Границы*, *Мёртвая зона*, *Максимальная скорость изменения* в режиме разработки можно вводить новые значения контроля. Если переменной назначен некоторый диапазон измерения, то в этих полях устанавливаются значения границ, унаследованные из настроек диапазона. В режиме исполнения, в качестве значений настроек, задаваемых на этой странице могут быть использованы текущие данные других переменных. Для этого нужно перетащить в нужное поле на данной странице ту переменную из дерева объектов, с помощью которой необходимо осуществлять контроль.

- ❖ **Важно!** *Переменная, данные которой планируется использовать для задания значения любого из параметров на этой странице должна принадлежать тому же компьютеру, что и настраиваемая переменная (для которой открыта данная страница свойств), иначе данные передаваться не будут!*

Кнопка *Вернуть умолчания* устанавливает значения полей в соответствии с выбранным диапазоном (или очищает поля, если диапазон не выбран).

Архивы

Существует два вида архивов: *Архивы данных* и *Архивы сообщений*. В MasterSCADA Архивы, как и документы, не существуют сами по себе - они принадлежат Объектам. При этом, никаких специальных действий по созданию архивов производить не нужно, поскольку любой объект, присутствующий в дереве объектов уже имеет в своем составе *Архив данных и сообщений*. Необходимость архивирования задается следующим образом:

- **Данные.** На странице свойств *Архив* переменной, данные которой необходимо записывать в архив, должен быть установлен флаг *Архивировать*;
- **Сообщения.** Необходимость архивирования задается не для отдельного сообщения, а для категории. Для того, чтобы сообщения какой-либо категории архивировались, необходимо для этой категории разрешить канал вывода *Архив*.

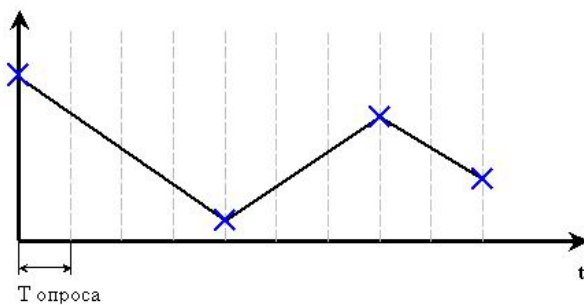
В режиме исполнения данные и сообщения от всех источников (переменных и ФБ), заносятся в *Архив объекта*, которому они принадлежат. Система архивирования в MasterSCADA устроена так, что запись данных в архив осуществляется только по изменению. Это позволяет избежать дублирования данных в архиве, и как следствие уменьшить его объем. Отличительной особенностью MasterSCADA является то, что при записи данных в архив используется не текущее время (на момент обработки в пакете), а именно метка времени архивируемой переменной, что позволяет получить более реальную картину. Это обстоятельство

является достаточно важным при работе с OPC серверами, которые сами устанавливают метки времени на присылаемые данные. Единственным требованием является монотонность меток времени данных по отдельным переменным. Другими словами, данные в Архив будут писаться только в том случае, если метка времени нового значения больше либо равна метке времени последнего записанного значения по данной переменной (вне зависимости от меток времени других переменных).

Для каждого из *Архивов Объекта*, можно установить максимально допустимый размер, а также длительность хранения (страница свойств объекта *Архив*). При этом, если в режиме исполнения срабатывает любое из этих ограничений, то перед записью новых данных, из архива удаляются наиболее старые записи. Данные и сообщения в архивах хранятся в двоичном формате. Размер архивной записи составляет 38 байт. Для просмотра архива данных следует экспортировать архив в файл *.mdb* (после чего с помощью Microsoft Access его можно просмотреть или конвертировать в удобный формат). Быстро просмотреть данные можно на закладке *Данные* у интересующей переменной (просмотр возможен, если у переменной установлен флаг *Архивировать*). Экспорт архивов данных или сообщений может осуществляться автоматически, по *Расписанию* объекта, или по действию *События*.

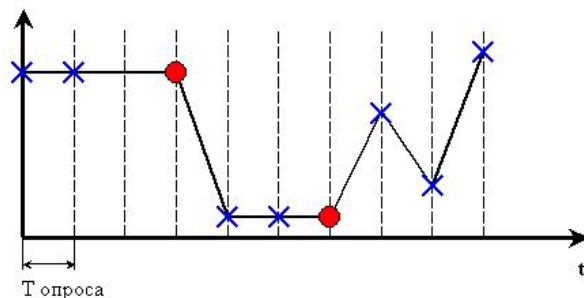
Архивирование данных

Во избежание дублирования данных, а также в целях оптимизации размеров архивов в MasterSCADA данные переменных архивируются по изменению. В некоторых случаях, возможны ситуации, когда значения в архиве будут не совсем корректно отображать тенденцию изменения данных. Например, если переменная изменяется достаточно редко (скажем раз в час), то несмотря на то, что физически параметр изменялся ступенчато, в архиве (и, соответственно, в тренде) мы можем увидеть картину указанную на рисунке ниже.

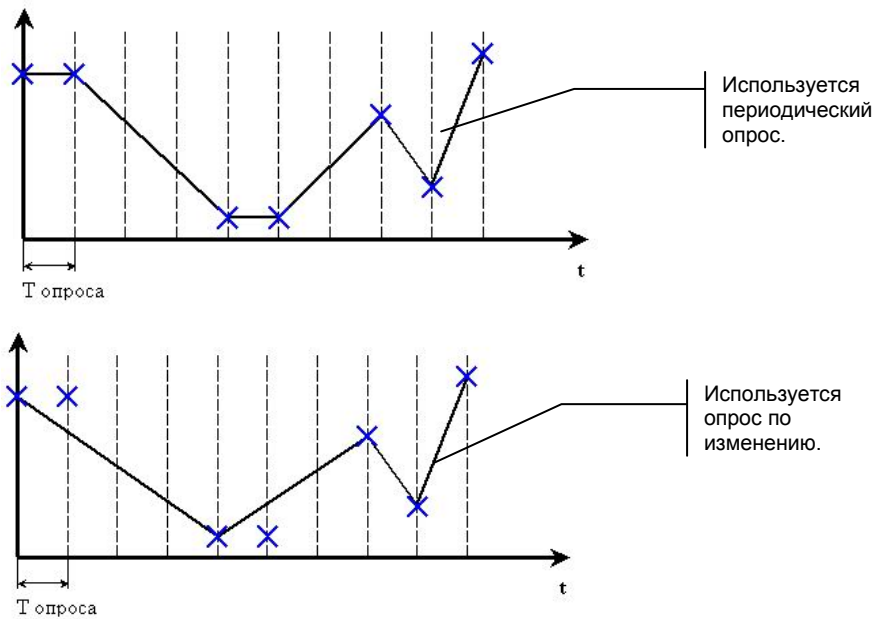


Естественно, это не совсем соответствует действительности. В MasterSCADA имеется специальная настройка, позволяющая приводить такие данные к ступенчатому виду. Эта настройка называется *Моделирование ступеньки*, она доступна со страниц свойств системы *Архив* на подзакладке *Шаблоны*.

Если флаг *Моделирование ступеньки* у переменной *установлен*, то перед записью нового значения в архив MasterSCADA дублирует предыдущее архивное значение с меткой времени, отстоящей на один период опроса от метки времени нового значения. Сказанное иллюстрирует приведенный ниже рисунок:



Синими крестиками показаны данные переменной, красными кружками - точки, доставляемые MasterSCADA. Следует отметить, что дополнительные точки не доставляются, если интервал между метками времени нового и последнего архивного значений не превышает двух периодов опроса. Если флаг *Моделирование ступеньки* у переменной *не установлен*, то дополнительные точки в архив не доставляются. При этом, в зависимости от способа опроса переменной возможны следующие ситуации:



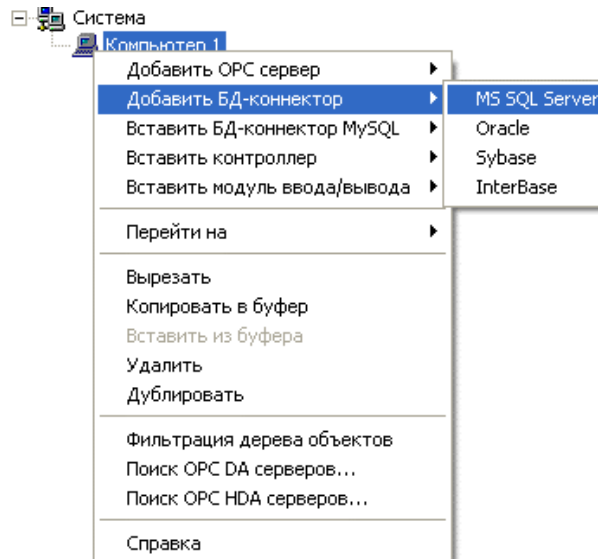
Синими крестиками обозначено изменение данных переменной, либо обновление метки времени данных на ее выходе.

Экспорт архивов

В MasterSCADA *Архивы данных* и *Архивы сообщений* хранятся и экспортируются отдельно. Экспорт архива в режиме разработки производится в модальном режиме (то есть другие функции системы не доступны пока длится процесс экспорта). В режиме исполнения это происходит в отдельном потоке и не мешает работе системы, однако занимает больше времени.

- **Пример:** Рассмотрим экспорт архивов на примере использования БД-коннектора MS SQL.

1. Добавьте в компьютер БД-коннектор MS SQL.



2. Если планируем экспортировать данные на верхний уровень, то на закладке *Использование БД-коннектора* необходимо поставить галочку *использовать для экспорта архивов*.

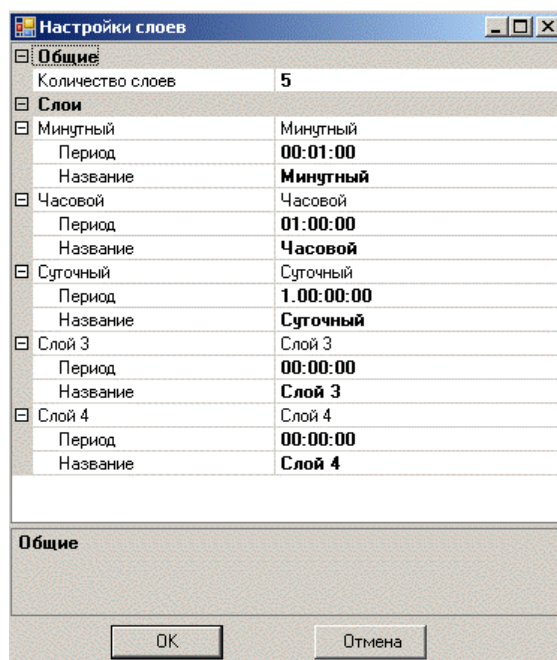
3. На странице свойств **Параметры БД-коннектора** задается: сервер (IP-адрес компьютера в сети, на котором стоит база данных), имя базы данных, подключение к БД (Аутентификация SQL сервера), имя пользователя (в БД должен быть создан этот пользователь и обладать правами администратора) и пароль.

В поле **Сервер** можно указать несколько ip-адресов (также как в сетевом взаимодействии). Разделитель ";" - отделяет ip, которые участвуют в переборе. Разделитель ":" - отделяет ip, которые не участвуют в переборе. В режиме разработки используется только первый ip. Если произошла ошибка подключения, то, при следующей попытке подключения будет использоваться следующий ip, кроме отделенных ":". В БД MS SQL передаются только значения. С сообщениями эта БД в MasterSCADA не работает.

4. Для конкретной переменной на закладке **Архив** указать экспорт в данный БД-коннектор. Если необходимо, чтобы все переменные попадали в БД-коннектор, то для Объекта на закладке **Архив** на подзакладке **Файловые архивы** указать экспорт значений в БД. В режиме исполнения данные будут попадать в БД согласно периоду опроса БД-коннектора. При экспортровании архива данных в БД создаются следующие таблицы: *Properties*, *Items*, *Data*. Описание таблиц и хранимых процедур имеется в Справке внутри программы MasterSCADA.

Формирование слоев

Слой данных – это отдельно хранимый набор архивных значений того же самого параметра, отличающийся периодом записи, способом предварительной обработки перед записью и длительностью хранения (в настоящее время реализованы только для архивов в БД). Основное назначение слоев – повышение быстродействия при извлечении данных для их отображения или обработки. Предполагается, что разработчик проекта заранее знает, их пользователь, например, будет интересоваться средние почасовые значения параметра или просмотр его ежеминутных текущих значений. Задав такие слои на этапе создания проекта, пользователь получит возможность при запросе этой информации для отображения на тренде или подготовке отчета извлечь все необходимые данные без поиска и обработки данных. Тем самым резко повышается быстродействие этих операций. По кнопке *Настроить слои...* на странице свойств **Архив** задается перечень слоев для системы в целом.



Сообщения

Сообщения - это информация для оператора, поступающая в текстовом виде. В MasterSCADA имеются различные способы отображения сообщений - *Каналы вывода* (принтер, окно сообщений, строка статуса, журнал сообщений, архив сообщений). В дальнейшем могут добавляться новые каналы (например, e-mail, факс, SMS-сообщения, речевой генератор). Сообщение всегда относится к той или иной *Категории*. Имеется ряд predefined категорий. Пользователь может создавать новые категории. *Категория* - атрибут, позволяющий сортировать и фильтровать сообщения в зависимости от их назначения.

Категория имеет имя и настройки:

- Цвет текста и фона сообщений;
- Звуковой файл, проигрываемый при возникновении сообщения;
- Приоритет;
- Список каналов вывода (для канала вывода *Журнал* задается перечень журналов, в которые будут поступать сообщения данной категории).

При просмотре журналов в режиме исполнения пользователь имеет возможность фильтрации и сортировки сообщений по выбранным категориям. *Приоритет* - атрибут, позволяющий сортировать и фильтровать сообщения в зависимости от их важности. Список приоритетов содержит четыре возможных величины: *Максимальный*, *Высокий*, *Средний*, *Обычный* - в порядке убывания (значения этих приоритетов в OPC стандарте соответственно 1000, 800, 500, 200 - то есть и максимальный и высокий приоритет относятся к наивысшему по стандарту приоритету при грубой разбивке диапазона на 3 части: до 333, до 667, до 1000). При просмотре журналов в режиме исполнения пользователь имеет возможность фильтрации и сортировки сообщений по выбранным приоритетам. Хотя приоритет и каналы вывода каждого сообщения устанавливаются по умолчанию от категории, которой принадлежит сообщение, они могут быть переопределены. Список журналов для каждого сообщения не переопределяется и используется список категории, которой принадлежит сообщение.

Типы сообщений

Сообщения бывают двух типов: *предопределенные* и *пользовательские* (событийные).

Предопределенные сообщения делятся на три основных группы:

- *Системные*, которые формирует сама программа, то есть система;
- *Сообщения функциональных блоков*, которые предусмотрел разработчик функционального блока и реализовал в нем;

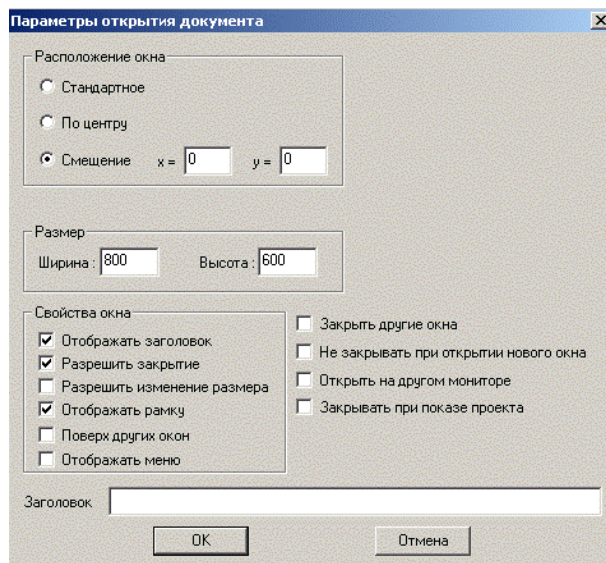
- *Состояния переменных* - сообщения, которые формируются автоматически в зависимости от значений выходов и их пограничных величин. Они делятся на три группы: *Авария*, *Предупреждение*, *Отсутствие значения*.


В каждой из этих групп находится несколько предопределенных сообщений. *Пользовательские сообщения* формируются по событиям. Они создаются при настройке *Событий* на закладке *Сообщение у События*.

Документы объектов

Настройка свойств открытия документа

Настройка параметров открытия окна производится с помощью диалогового окна, показано ниже:



Для вызова этого диалогового окна, следует перейти на закладку *Окна* и нажать кнопку  у соответствующего документа. Группа *Расположение окна* - становится доступна, только в том случае, если для документа установлен способ открытия *Окно* или *Новое окно*.

- *Стандартное* - при открытии окно располагается в левом верхнем углу экрана;
- *По центру* - окно располагается по центру экрана;
- *Смещение* - позволяет для открываемого окна задать смещение в пикселях относительно левого верхнего угла экрана.

Группа *Размер* - доступна только для трендов и журналов. Размеры мнемосхем задаются на странице свойств системы *Окна*, и могут быть переопределены у каждого компьютера. Группа *Свойств Окна* - становится доступна, только в том случае, если для документа установлен способ открытия *Окно* или *Новое окно*.

- Флаг *Отображать заголовок* - при открытии окна отображается заголовок;
- Флаг *Разрешить закрытие* - если флаг не стоит, то окно закроется только в случае остановки проекта;
- Флаг *Разрешить изменение размера* - позволяет в режиме исполнения изменять размер окна;
- Флаг *Отображать рамку* - если флаг установлен, то в режиме исполнения отображается рамка объекта;
- Флаг *Поверх других окон* - при открытии окно располагается поверх других окон;
- Флаг *Закрывать другие окна* - когда установлен, при открытии текущего окна закрываются все ранее открытые окна;
- Флаг *Не закрывать при открытии нового окна* - когда установлен, при открытии текущего окна не закрываются все ранее открытые окна;
- Флаг *Открыть на другом мониторе* - если кнопка вызова (кнопка перехода к объекту) находится на мнемосхеме, открытой как *Окно* на мониторе 1, и если способ вызова по кнопке перехода задан *Новое окно*, то новая мнемосхема по этой кнопке откроется на мониторе 2;

- Флаг *Закрывать при показе проекта* - окно, открытое с этой настройкой закрывается при нажатии *Показать проект*;
- Флаг *Включить имя экземпляра в заголовок* - в заголовке окна отображается имя экземпляра. В поле *Заголовок* вводится название окна.

Существует три способа открытия документов в режиме исполнения:

- *Документ* - открывается в окне MasterSCADA вместо страниц свойств (как в режиме разработки);
- *Окно* - открывается в отдельном окне (в этом случае для документов будет одно отдельное окно, в котором их можно будет открывать по очереди);
- *Новое окно* - открывается в новом отдельном окне (в отличие от просто окна в этом случае можно открыть одновременно несколько документов. Для каждого документа, которому назначен такой способ открытия будет создаваться отдельное окно).

Способ открытия для каждого типа документа, установленный в настройках системы или *Компьютера* (страница свойств *Окна*) наследуется всеми подчиненными объектами. При этом, выполняются следующие условия:

- *Документ* - для документа данного типа у всех подчиненных объектов устанавливается способ открытия *Документ*, изменить который нельзя;
- *Окно* - для документа данного типа у всех подчиненных объектов существует возможность выбора способа открытия *Окно* или *Новое окно*. По умолчанию устанавливается *Окно*;
- *По выбору* - для документа данного типа у всех подчиненных объектов существует возможность выбора способа открытия: *Документ*, *Окно* или *Новое окно*. По умолчанию устанавливается *Документ*.

Документы мнемосхемы

Объект может иметь стандартные документы, которые открываются в редакторе MasterGraph. Документ каждого типа (например, мнемосхема) может быть у объекта только один, но типов документов может быть несколько, (включая предопределенные типы). Таким образом, если нужны три мнемосхемы для одного объекта, создайте еще типы *Мнемосхема 2*, *Мнемосхема 3* и по одному документу на каждый. Типы документов создаются и настраиваются сразу для всех объектов на закладке *Окна у Системы*, и могут быть изменены для каждого *Компьютера*. После настройки типов документов (окон) в системе, создаются документы для *Объектов*. Тиражирование объектов происходит вместе с его документами. При этом работает механизм восстановления внешних связей с другими объектами.

Рапорты объектов

Рапорты используются для просмотра и печати данных в определенный момент времени. Объект может иметь несколько рапортов. Для создания рапортов используется Microsoft Excel, который открывается прямо в MasterSCADA во внедренном режиме. Возможно сконфигурировать необходимую таблицу и перетаскать мышью переменные из дерева объектов в ячейки. По правой кнопке мыши перетаскиваются названия перетаскиваемых элементов дерева объектов. В режиме исполнения туда будут помещаться текущие данные, то есть значения на данный момент времени. В режиме исполнения возможно просматривать и печатать рапорты. Эти действия так же удобно автоматически осуществлять в заданное время с помощью расписания объекта.

Создание кнопок переходов

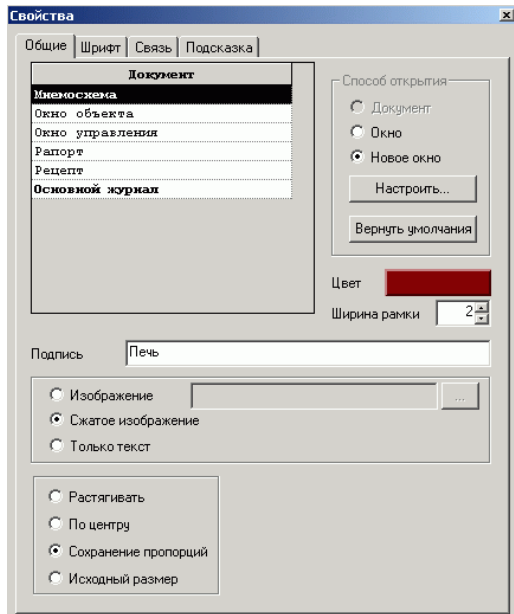
Объекты и *Компьютеры* можно перетаскивать мышью в окна стандартных документов (мнемосхем), причем как в собственную, так и в мнемосхемы других объектов. При этом в мнемосхеме создается кнопка, по которой в режиме исполнения можно осуществить переход к любому документу перетаскиваемого объекта. На кнопку может быть помещено сжатое изображение выбранной мнемосхемы, точечный рисунок, либо статический текст. Если перетаскивать *Объект* или *Компьютер*, держа нажатой левую кнопку, то документ, который будет открываться по кнопке объекта выбирается следующим образом:

- **Для Компьютеров.** Если компьютер имеет стартовую мнемосхему, то создается кнопка перехода на стартовую мнемосхему компьютера. Если стартовой мнемосхе-

мы у перетаскиваемого компьютера нет, то создается кнопка перехода к первому из списка имеющихся у него документов;

- **Для Объектов.** Создается кнопка для перехода к документу по умолчанию, который устанавливается для *Объекта* на его странице свойств *Окна*.

При перетаскивании *Объекта* или *Компьютера* правой кнопкой мыши, в появляющемся контекстном меню из списка доступных документов можно выбрать тот, который будет открываться при нажатии на кнопку в режиме исполнения. По умолчанию кнопка перехода к документам наследует способы открытия документов из настроек *Объекта* или *Компьютера*. Эти настройки, а также тип открываемого по кнопке документа могут быть переопределены на ее странице свойств *Общие*, внешний вид которой показан на рисунке ниже.



Список документов *Объекта/Компьютера*. Жирным шрифтом отображаются уже созданные документы. Тот документ, который выделен в списке (черным фоном) и будет открываться при нажатии на кнопку в режиме исполнения. Поле *Экземпляр* позволяет выбрать экземпляр типового объекта который будет вызывать кнопка. *По выбору* - при нажатии будет вызываться список со всеми экземплярами, из которого можно выбрать нужный. *Все* - окно не привязано к конкретному экземпляру, а отображает значения текущего экземпляра. Если настройка включена для журнала, то в этом случае в журнале отображаются сообщения от всех экземпляров. Флаг *Добавлять в подпись* - в подпись кнопки будет добавлено имя экземпляра. Остальные настройки определяют внешний вид кнопки.

Генератор отчетов

Генератор отчётов интегрируется в MasterSCADA как редактор документов. Для создания нового отчёта нужно на странице свойств объекта *Отчёты* нажать кнопку *Добавить отчёт*. Для каждого отчёта в папке объекта создаётся своя папка, содержащая несколько файлов. Отчёт MasterSCADA - это шаблон, на основании которого генерируется непосредственно реальный отчёт, в котором отображаются данные из MasterSCADA и других источников. Отчёт можно редактировать (кнопка *Редактировать*), просматривать - при это будет сгенерирован реальный отчёт и отображен в отдельном окне (кнопка *Просмотреть*), сохранить (кнопка *Сохранить*) и распечатать (кнопка *Печать*).

В режиме разработки при просмотре отчёт отображается в основном интерфейсе MasterSCADA, в режиме исполнения отображается в окне поверх других. В режиме разработки при печати будет отображено окно выбора принтера, в режиме исполнения печать направляется на принтер по умолчанию.

Для использования мгновенных значений из MasterSCADA осуществляется привязка переменных MasterSCADA к переменным отчёта. Для использования архивных значений используется ФБ *Архивные данные для рапорта*, представленный в генераторе отчётов в виде *Источника данных Архивные ФБ*.

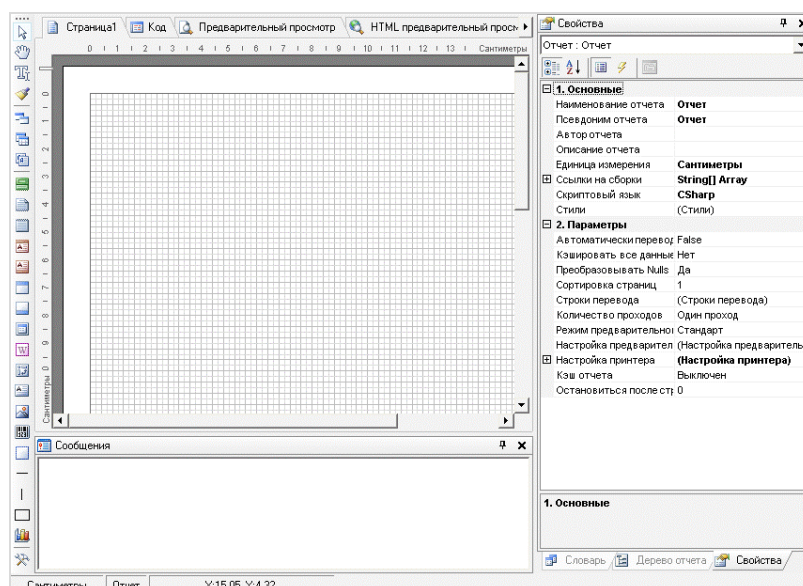
Переменные и архивные ФБ можно связывать с отчётом с помощью Drag&Drop. Перетаскивать можно как на страницу отчёта, так и в словарь отчёта. При перетаскивании на страницу

создаётся элемент на странице, который связывается с элементом в словаре, представляющим перетаскиваемую сущность MasterSCADA. Если в словаре нет элемента, который связан с элементом MasterSCADA, он автоматически создается. При перетаскивании в словарь учитывается элемент, на который осуществляется перетаскивание; если к этому элементу словаря можно привязать перетаскиваемый элемент MasterSCADA, то происходит перепривязка, в противном случае - если в словаре нет элемента, связанного с перетаскиваемой сущностью, то он создается.

Функциональные блоки MasterSCADA связываются с источником данных в отчёте. У источника данных есть имя, псевдоним и имя в источнике. Псевдоним источника можно менять по своему усмотрению. Имя источника не рекомендуется менять. Имя в источнике менять запрещено (через него осуществляется связь с ФБ MasterSCADA). У столбцов источника данных также есть имя, псевдоним и имя в источнике. На столбцы распространяются те же правила, что и на источники.

Переменные MasterSCADA связываются с переменными генератора отчётов. У переменных генератора есть имя и псевдоним. Имя переменной отчета, связанной с переменной MasterSCADA не должно изменяться. Псевдоним можно менять по своему усмотрению. При привязке, если псевдоним не был изменён пользователем, он будет автоматически изменён на новый, созданный на основе имени переменной MasterSCADA. Переменные отчёта, связанные с переменными MasterSCADA, являются функциями. Если у переменной MasterSCADA есть значение до опроса или значение восстановления после рестарта, то данное значение будет доступно в режиме разработки.

Окно отчёта изображено на рисунке ниже.



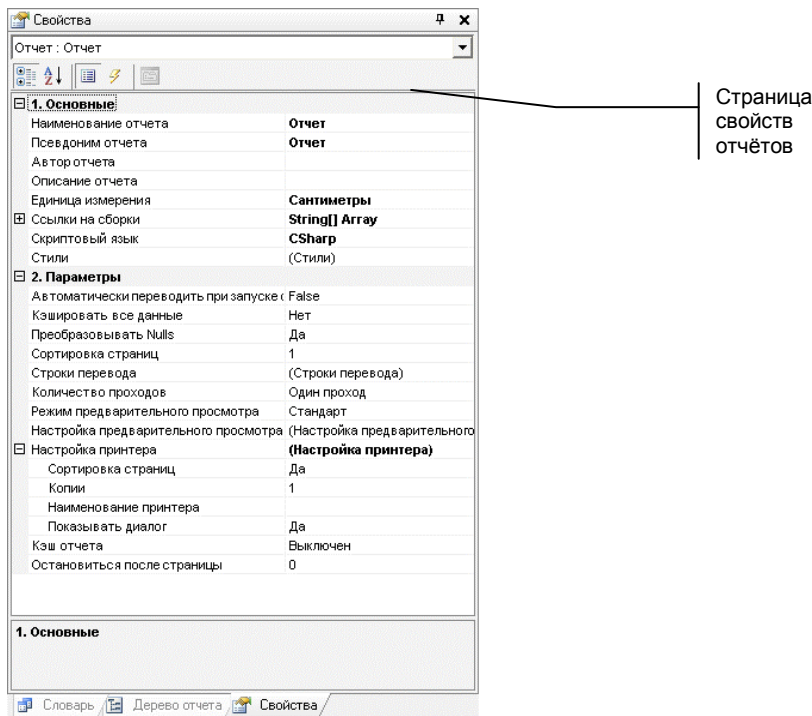
В генераторе отчетов существует три понятия отчетов:

- *Шаблон отчета* - отчет с привязкой к переменным MasterSCADA;
- *Прототип отчета* - отчет без привязки к переменным MasterSCADA;
- *Экземпляр отчета* - конечный вид отчета.

Прототип отчета делится на страницы. Каждая страница может иметь свои размеры и поля печати. Все компоненты в отчете располагаются на страницах. При построении отчета генератор последовательно обрабатывает все страницы отчета. Отчеты строятся из компонентов. Компонент - это объект, который размещается на странице. Компоненты обладают самой разной функциональностью. Все компоненты можно разделить на пять основных групп:

- Страницы;
- Бэнды;
- Контейнеры;
- Компоненты для отображения информации;
- Элементы управления.

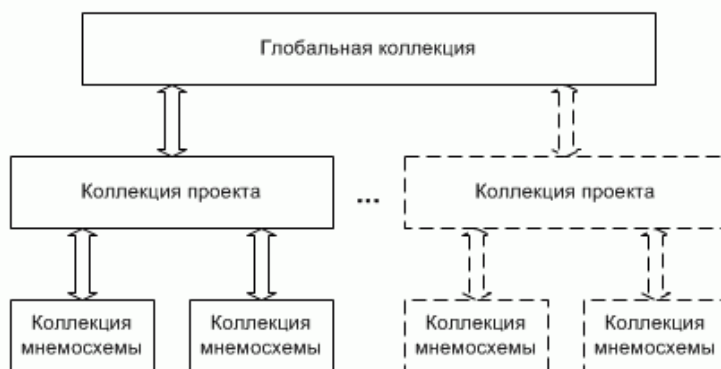
Комбинация компонентов из этих групп и образует отчет. Компонент обладает набором свойств, событий и методов. Изменяя свойства и задавая события компонента, можно изменить поведение компонента.



Управление ресурсами

Коллекции ресурсов

Под ресурсами понимаются внешние файлы (графические, звуковые), используемые в проектах и элементами мнемосхем, а также пользовательские мультфильмы, создаваемые в редакторе мнемосхем MasterGraph. Коллекции представляют собой специальные папки для хранения таких файлов. При вставке внешних файлов в проект, или в элемент мнемосхемы, они копируются в соответствующие коллекции, что позволяет обеспечить легкую переносимость проектов и их элементов. Мультфильмы и их настройки хранятся в отдельных файлах, имеющих расширение *.fmc*. Такие файлы имеются в коллекциях каждого уровня. Схема организации коллекций представлена ниже:



Коллекции мнемосхем. Каждая мнемосхема имеет свою собственную коллекцию ресурсов, расположенную в папке *Res*, находящейся в каталоге мнемосхемы. Эта коллекция содержит графические файлы, используемые элементами данной мнемосхемы. Файлы ресурсов помещаются в эту коллекцию автоматически, как только вставляется внешний файл изображения в какой-либо элемент мнемосхемы. Ресурсы из коллекции мнемосхемы доступны только для элементов этой мнемосхемы. Для хранения мультфильмов мнемосхемы, в коллекции мнемосхемы содержится специальный файл, имеющий имя *Мои мультфильмы.fmc*.

Коллекция проекта. Каждый проект имеет свою собственную коллекцию ресурсов, расположенную в папке *Res*, которая находится в каталоге проекта. Коллекция проекта содержит все файлы ресурсов, используемые на уровне проекта. Файлы ресурсов помещаются в эту коллекцию автоматически, как только в проекте начинают использоваться внешние файлы

(например, файл для стартовой заставки компьютера). Файлы из проектной коллекции доступны любым элементам проекта, а также элементам всех мнемосхем, входящих в этот проект. Кроме того, для хранения мультфильмов проекта, в коллекции существует специальный файл, имеющий имя *Проект.fmc*.

Глобальная коллекция. Физически глобальная коллекция расположена в папке *Pictures*, которая находится в каталоге проектов MasterSCADA (имя по умолчанию - *MasterSCADA Projects*). В этой коллекции находятся файлы ресурсов, доступные из любых проектов и мнемосхем. Кроме того, в глобальной коллекции имеется файл с мультфильмами (*MasterSCADA.fmc*), содержимое которого доступно для любой мнемосхемы любого проекта.

Файлы, содержащие мультфильмы имеют строго определенные имена, и переименовывать их нельзя. При создании пользовательского мультфильма в мнемосхеме возможно использовать мультфильмы из любой коллекции, пользуясь соответствующими кнопками в диалоге настройки. Элементы проекта и мнемосхем при выборе внешних ресурсов запоминают только имена файлов (т.к. эти файлы копируются в соответствующие коллекции), и далее в процессе работы, производят их поиск в коллекциях ресурсов снизу вверх, начиная с текущей коллекции элемента: если нужный файл в текущей коллекции не был обнаружен, то осуществляется поиск в коллекции более высокого уровня и т.д. Благодаря такому механизму, существует возможность размещать общие файлы для нескольких элементов в коллекциях более высокого уровня, что дает возможность несколько уменьшить суммарный объем проекта, а также редактировать общие файлы всего лишь в одном месте. Манипулировать содержимым коллекций рекомендуется с помощью Органайзера ресурсов.

Перемещение ресурсов между коллекциями

В процессе работы с проектом возможно перемещать ресурсы между коллекциями мнемосхем, проекта и глобальной коллекцией вручную, с помощью *Органайзера ресурсов*. Но в некоторых ситуациях, для того чтобы обеспечить правильную работу элементов проекта MasterSCADA делает это автоматически:

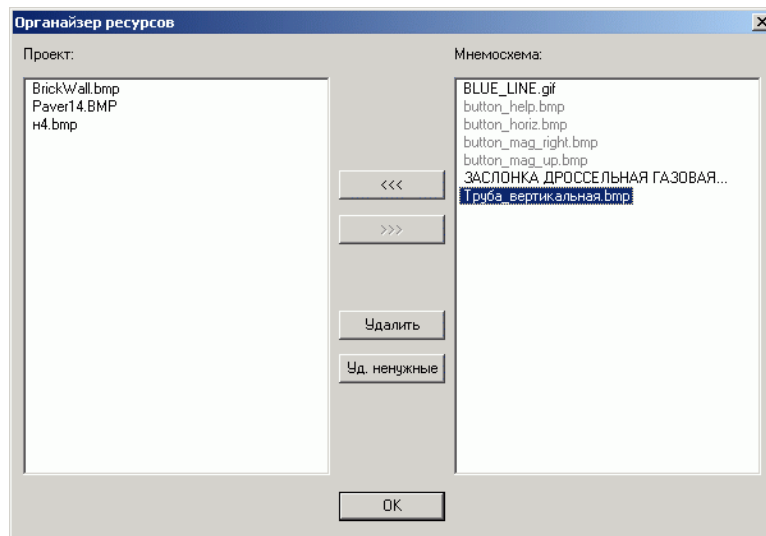
- **При экспорте проекта.** Для всех элементов проекта (включая элементы всех мнемосхем, входящих в состав проекта) анализируются используемые ими ресурсы. Если обнаруживается, что элемент использует ресурсы из глобальной коллекции, то этот файл копируется в проектную коллекцию нового (проэкспортированного) проекта. Если в какой-либо из мнемосхем используется мультфильм из глобальной коллекции, то он копируется в коллекцию соответствующей мнемосхемы в новом проекте;
 - **При экспорте мнемосхемы.** Для всех элементов мнемосхемы производится анализ используемых ими ресурсов. Если обнаруживается, что элемент использует файл из коллекции проекта, или из глобальной коллекции, то этот файл копируется в коллекцию проэкспортированной мнемосхемы. Кроме того, в коллекцию мнемосхемы копируются все внешние мультфильмы (из глобальной и проектной коллекции);
 - **При копировании, дублировании и помещении объекта в библиотеку.** Для всех элементов всех мнемосхем объекта производится анализ используемых ими ресурсов. Если обнаруживается, что некоторый элемент использует файл из коллекции проекта, или из глобальной коллекции, то этот файл копируется в ресурсы соответствующей мнемосхемы нового объекта. Помимо этого, в коллекции мнемосхем копируются все внешние мультфильмы (из глобальной и проектной коллекции);
 - **При помещении элемента мнемосхемы в библиотеку:** Файлы, используемые элементом (вне зависимости от коллекции, в которой они располагаются) помещаются в глобальную коллекцию ресурсов. Если этот элемент является мультфильмом, то его настройки копируются в глобальную коллекцию.
- ❖ **Важно!** При перемещении мультфильма в коллекцию, в которой уже имеется мультфильм с таким именем, копирования не происходит.

Органайзер ресурсов

Запуск **Органайзера ресурсов** из проекта осуществляется командой меню Сервис-Органайзер-Ресурсы, из редактора мнемосхем: Сервис-Органайзер Ресурсы.

С помощью органайзера ресурсов возможно манипулировать содержимым всех трех коллекций ресурсов - перемещать отдельные файлы между коллекциями, просматривать неиспользуемые файлы, удалять файлы и т.д. Органайзер ресурсов может быть открыт как из

редактора мнемосхем (при этом в левом окне показывается содержимое коллекции проекта, а правом - коллекции текущей мнемосхемы), так и из среды разработки проекта MasterSCADA (в этом случае в левом окне показывается содержимое глобальной коллекции, в правом - коллекции проекта). Внешний вид Органайзера ресурсов показан на рисунке ниже.



Серым цветом отмечены неиспользуемые файлы, т.е. те на которые не ссылается ни один элемент. С помощью кнопок <<< и >>> можно перемещать файлы ресурсов между коллекцией проекта и мнемосхемы. Кнопка *Удалить ненужные* позволяет удалить из ресурсов мнемосхемы (или проекта) неиспользуемые файлы.

Автоматическая установка периодического опроса

По сравнению с периодическим опросом, опрос по изменению является более эффективным, поскольку функциональные блоки и переменные пересчитываются не на каждом такте цикла вычислений, а только тогда, когда они действительно требуют пересчета (изменились данные хотя бы одного входа ФБ с момента последнего пересчета). Не все функциональные блоки и переменные могут корректно работать при опросе по изменению входов (например те, которые входов не имеют). Поэтому, при вставке таких ФБ в дерево объектов, для них автоматически устанавливается периодический способ опроса.

Список ФБ, требующих периодического опроса приведен ниже:

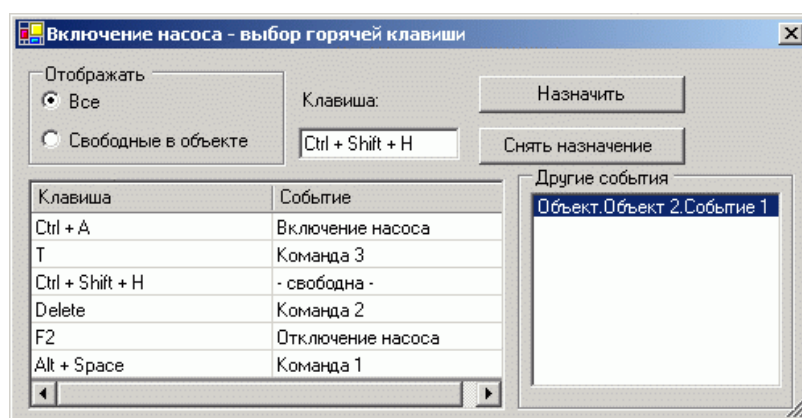
Категория	Функциональный блок
Управление	Кнопка. Регулятор. Циклограмма.
Работа с архивом	Мгновенное значение по архиву. Среднее значение по архиву. Интеграл по архиву. Счетчик импульсов по архиву. Счетчик пробега по архиву.
Алгоритмы VNS / Генераторы	Пила. Пульсатор. Шум.
Алгоритмы VNS / Обработка аналоговых параметров	Звено запаздывания.
Алгоритмы VNS / Время	Выдержка времени. Импульс по времени.
Служебные	Текущее время. Служебная информация.
Другие	Тестовый генератор.

Кроме того, периодический опрос устанавливается для *Расчета / События*, если:

- В формуле используются функции времени без параметров: Time, Date, DateTime, CurHour, CurShift, CurWeek, CurDay, CurMonth, CurYear;
- В формуле используются функции для работы с архивом: ArchIntegral, ArchAverage, ArchDiscreteCounter, ArchDiscreteCounterTime, ArchMomentValue;
- Расчет не содержит параметров (результат вычисления формулы - константа).

Назначение горячих клавиш

MasterSCADA в режиме исполнения позволяет использовать горячие клавиши при работе с переменными типа *Событие* и *Команда дискретного типа*. При нажатии комбинации из одной или нескольких клавиш происходят действия, назначенные для того или иного события или команды. Действия назначаются на странице свойств *Действия* у событий и дискретных команд). Настройка горячих клавиш производится в режиме разработки. На странице свойств *Общие* вышеуказанных переменных появляется область, в которой располагается текстовое поле с названием горячих клавиш для этой переменной и кнопка *Выбрать*. По кнопке *Выбрать* открывается диалоговое окно, в котором производится настройка горячих клавиш.



Функции элементов управления:

- Поле *Клавиша* определяет комбинацию горячих клавиш. Горячей клавишей может выступать любая клавиша клавиатуры, за исключением клавиши *Tab*. Клавиши *Ctrl*, *Shift*, *Alt* могут использоваться только в комбинации с другими клавишами;
- Кнопка *Назначить* связывает событие с комбинацией клавиш, отображенной в поле *Клавиша* и закрывает диалог;
- Кнопка *Снять назначение* разрывает связь между событием и установленной для него ранее комбинацией клавиш и закрывает диалог. Таблица включает в себя два столбца:
 - В столбце *Клавиша* отображаются имена всех клавиш, которые уже используются в проекте;
 - В столбце *Событие* отображаются имена переменных задействованных в данном объекте, связанные с этими клавишами. В случае, если горячие клавиши установлены для переменных, находящихся в других объектах проекта, то в столбце *Событие* появляется надпись *-свободна-*. Это означает, что данная комбинация может быть использована для данной переменной;
- Поле *Другие события* отображает полное имя переменных, расположенных в других объектах и имеющих горячую клавишу, которая в данный момент выделена;
- Группа *Отображать* определяет, какие переменные будут отображаться в таблице:
 - В случае, если отмечена кнопка *Все*, то в таблице находятся все переменные, для которых назначены горячие клавиши;
 - В случае если отмечена кнопка *Свободные в объекте*, то отображаются только переменные принадлежащие другим объектам проекта.

Назначить горячую клавишу *Событию* или *Дискретной команде* можно одним из следующих способов: выделить в таблице свободную клавишу и нажать кнопку *Назначить* или установить курсор в поле ввода *Клавиша*, нажать на клавиатуре нужную комбинацию и нажать кнопку *Назначить*. Используется в случае, если комбинация еще нигде не назначена.

В режиме исполнения, при нажатии горячей клавиши формируется событие, связанное с ней, для того объекта, который выбран сейчас в проекте, либо с документом, с которым работает в настоящее время пользователь. Если такового нет, то поиск производится вверх по иерархии вплоть до корневого узла дерева *Объектов*.

Подсчет количества связей

Подсчет количества связей и задействованных выходов ФБ производится отдельно для каждого компьютера системы по следующим категориям (подсчет каждого раздела производится независимо, они не суммируются – ограничение срабатывает по тому разделу, в котором количество первым достигнет установленного значения максимума):

Связи с OPC переменными. Подсчитывается количество всех переменных всех OPC серверов принадлежащих данному компьютеру, вне зависимости от того используются данные этих переменных или нет. Таким образом, если OPC переменная присутствует в дереве системы, то связь считается установленной. Помимо этого также учитываются задействованные входы-выходы контроллера, а также входы-выходы поддерживаемых плат. Вход-выход считается задействованным, если он имеет связь с каким-либо элементом проекта или у которого установлен флаг *Архивировать*.

Связи с другими компьютерами. Связь компьютера А с компьютером В считается установленной (и подсчитывается для обоих компьютеров А и В) в следующих случаях:

- Если переменная OPC сервера, принадлежащего компьютеру А, имеет потребителя, исполняемого на компьютере В;
- Если объект, принадлежащий компьютеру А, имеет в своем составе переменную - выход (*Команду, Расчет* или *Событие*), которая имеет потребителя, исполняемого на компьютере В;
- Если *Выход функционального блока*, исполняемого на компьютере А имеет потребителя, исполняемого на компьютере В;
- Если *Выход документа* (мнемосхемы или тренда) объекта, исполняемого на компьютере А имеет потребителя, исполняемого на компьютере В.

Потребителем в данном случае может быть *переменная MasterSCADA (Значение или Команда), Вход функционального блока, Вход документа* (мнемосхемы или тренда), *OPC переменная для записи*. Открытие мнемосхемы или тренда в режиме исполнения может привести к образованию дополнительных межкомпьютерных связей, которые не были подсчитаны в режиме разработки. Эти связи подсчитываются в режиме исполнения. В зависимости от того, какому компьютеру принадлежит открываемый документ, возможны следующие ситуации:

- Открывается мнемосхема или тренд, принадлежащие другому компьютеру. В этом случае дополнительно подсчитываются следующие связи: *Связи между ФБ, исполняемым на другом компьютере, и его представлением в мнемосхеме* (визуальные выходы, количество визуальных выходов зависит от конкретного ФБ), *Связи входов/выходов мнемосхемы, с переменными, исполняемыми на других компьютерах.* (количество связей определяется количеством входов/выходов);
- Открывается мнемосхема, принадлежащая данному компьютеру. В этом случае дополнительно подсчитываются следующие связи: *Связи между ФБ, исполняемым на другом компьютере, и его представлением в мнемосхеме* (визуальные выходы). Количество визуальных выходов зависит от конкретного ФБ.

Если действительное количество межкомпьютерных связей в режиме исполнения превысит максимально допустимое (в зависимости от конфигурации пакета), то будет выдано предупреждающее сообщение, и новые связи не установятся.

Количество задействованных выходов функциональных блоков. Начиная с версии 3.5 выходы библиотечных функциональных блоков при подсчете лицензионных связей не учитываются – лицензионными точками считаются только пользовательских ФБ и ВФБ (то есть функциональных блоков, написанных самим пользователем), а также выходы модуля *Скрипт C#, MySQL Процедура/функция, Сохранение в файле, Сохранение в двоичный файл, MasterPLC SmartLink*.

Выход функционального блока считается задействованным, если он имеет потребителя или данные, которые он получает архивируются. В связи с тем, что некоторые связи выхода ФБ

устанавливаются только в режиме исполнения, подсчет количества задействованных выходов ФБ производится не только в режиме разработки, но и в режиме исполнения. В частности, связи выходов ФБ с рапортами и с мнемосхемами (в случае, если выход ФБ перетасчен в мнемосхему) определяются только в режиме исполнения. Таким образом, выход ФБ считается задействованным в следующих случаях:

- Если данные этого выхода архивируются (на странице свойств *Общие* выхода ФБ установлен флаг *Архивировать*);
- Если выход имеет потребителя. Потребителем в данном случае может быть любая переменная - вход или вход документа (мнемосхемы или тренда);
- Если выход перетасчен в ячейку рапорта, или в мнемосхему в виде контрола (определяется в момент открытия документа в режиме исполнения).

Для визуальных функциональных блоков отдельно подсчитывается также количество визуальных выходов. Визуальный выход необходим для связи ФБ с его представлением в мнемосхеме. Визуальные выходы ВФБ защиты в самом функциональном блоке, и считаются задействованными всегда, даже если ВФБ не представлен в мнемосхеме.


В том случае, если выход ФБ имеет несколько потребителей (переменных, входов документов, рапортов и т.д.), то он считается один раз. Если действительное количество задействованных выходов ФБ в режиме исполнения превысит максимально допустимое (в зависимости от конфигурации пакета), то будет выдано предупреждающее сообщение, и новые связи не установятся. Если функциональный блок находится в объекте, который исполняется в контроллере, и не имеет связей с переменными объектов, исполняемых на компьютере, не используются на мнемосхемах и в других документах MasterSCADA, то тогда выходы этих функциональных блоков в подсчете количества связей не учувствуют. Если подобные связи у ФБ есть, то они подсчитываются также как если бы объект исполнялся на компьютере.


Связи с БД. Считаются только задействованные входы-выходы функций и хранимых процедур. Вход-выход считается задействованным, если он имеет связь с каким-либо элементом проекта или у которого установлен флаг *Архивировать*.


Входы выхода контроллера. Подсчитываются задействованные входы и выходы контроллеров. Вход-выход считается задействованным, если он имеет связь с каким-либо элементом проекта или у которого установлен флаг *Архивировать*.

Работа в режиме исполнения

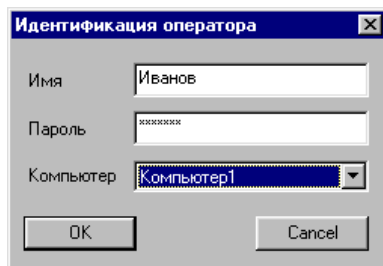
Существует три возможных варианта работы в режиме исполнения.

Рабочий режим. Включается командой *Пуск* из меню *Режим*, или кнопкой  на панели инструментов. Это основной режим исполнения. В рабочем режиме должен быть осуществлен переход к нему на всех компьютерах системы. Программа производит реальное управление технологическим процессом. Рабочий режим может быть также запущен из командной строки: *MasterSCADA.exe <имя проекта>* или *MasterSCADA_RT.exe <имя проекта>*. (<имя проекта> может быть названием проекта, если проект находится в стандартной папке *Projects* или полным именем файла проекта, *.vav с путем к нему).

Режим отладки. Включается командой *Отладка* из меню *Режим*, или кнопкой  на панели инструментов. Этот режим предназначен для отладки проекта на одном компьютере. Независимо от того, сколько компьютеров находится в дереве системы, все объекты, функциональные блоки, OPC серверы создаются на текущем компьютере и все действия производятся на нем.

Режим имитации. Включается командой *Имитация* из меню *Режим*, или кнопкой  на панели инструментов. В этом режиме на все входы, не имеющие связей вместо констант будет подаваться имитация в соответствии с настройками Системы.

При переходе в режим исполнения на компьютере создаются для работы все элементы, вставленные в деревья, для чего может понадобиться некоторое время. При старте режима отладки все элементы создаются на текущем компьютере. При старте основного режима исполнения сначала выводится диалог.



В диалоге выбирается имя компьютера, за который будет работать текущий. Для распределенной работы нужно просто скопировать проект на все компьютеры сети, запустить режим исполнения и выбрать на компьютерах разные имена. Все остальное происходит автоматически.

При запуске проекта из командной строки MasterSCADA автоматически загружает указанный проект и переходит в режим исполнения. Синтаксис командной строки имеет вид: *<путь к MasterSCADA.exe> <путь к файлу проекта> <параметры>*

- *Путь к MasterSCADA.exe* - может быть например, таким "C:\Program Files\Master SCADA\MasterSCADA.exe";
- *Путь к файлу проекта* - например, "D:\Master SCADA Projects\Projects\ Электросети\Электросети.vav";
- *Параметры* - с помощью параметров указывается за какой компьютер производится запуск проекта, основной компьютер или резервный, и задается имя пользователя и пароль.

Доступны следующие параметры:

- */computer:<логическое имя компьютера>* - Если в проекте имеется только один компьютер, то этот параметр можно не использовать;
- */masterstatus:0* - резервный компьютер - Если в проекте не используется резервирование, то данный параметр указывать не нужно;
- */masterstatus:1* - основной компьютер - Если в проекте не используется резервирование, то данный параметр указывать не нужно;
- */user:<имя пользователя>* - Обязательный параметр;
- */password:<пароль>* - Пароль оператора. Если для оператора пароль не назначен, то этот параметр можно не указывать;

- `/aero` - Не отключает оформление рабочего стола;
- `/Logo:<C:\Имяфайла.bmp>` - Заменяет заставку на пользовательскую. Задается полное имя файла в формате bmp;
- `/ImportProjectFrom:192.168.1.2` - MasterSCADA подключится к серверу с данным адресом, скачает с него текущую работающую версию проекта (если на данном компьютере она уже есть, то обновит только новые файлы) и запустит проект. Если подключиться не удастся, или обновление завершится с ошибками то MasterSCADA закроется. Аналогично автоматической синхронизации проекта по старту;
- `имя_проекта" /ImportProjectFrom:<адрес>` - Перед скачиванием проекта будет произведена проверка по имени проекта. Если имя запущенного на сервере проекта отличается от `имя_проекта`, то загрузка произведена не будет. Используется только в сетевом проекте.

С помощью приведенной ниже команды запускается проект `Электросети.vav` за компьютером `Диспетчерская` оператором `Иванов`, имеющим пароль `12345`:

```
"C:\Program Files\InSAT\Master SCADA\MasterSCADA.exe" "D:\Master SCADA
Projects\Projects\Электросети\Электросети.vav" "/computer:Диспетчерская" "/user:Иванов"
"/password:12345"
```

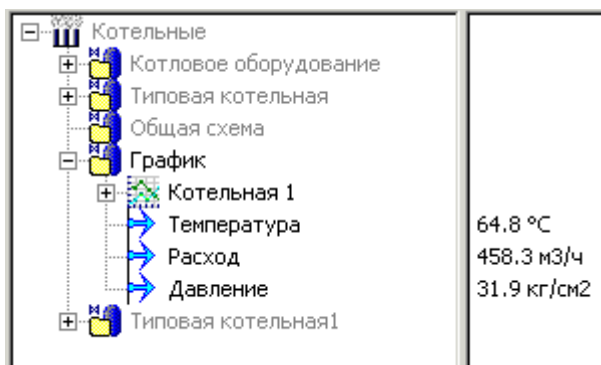
Обратите внимание, что все параметры записаны в кавычках. Для запуска проекта по ярлыку необходимо в ярлыке прописать строку, указанную выше.

- ❖ **Важно!** Для запуска MasterSCADA при старте операционной системы необходимо получившийся ярлык поместить в **Пуск/Все программы/Автозагрузка**.

Пользовательский интерфейс в режиме исполнения

Дерево объектов (RT)

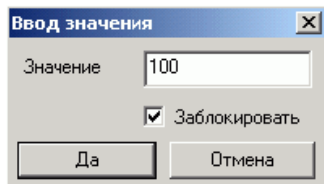
В режиме исполнения **Дерево Объектов** принимает вид:



Окно **Дерева Объектов** разделяется на две части вертикальной чертой. В левой части можно видеть само дерево объектов в том же виде, в каком оно было в режиме разработки. Объекты, не принадлежащие данному компьютеру, выделяются в дереве серым шрифтом. Справа от разделительной черты отображаются текущие значения переменных.

- ❖ **Важно!** Это поле обновляется периодически, поэтому для данных, изменяющихся быстро и циклически возможны некоторые расхождения с тем, что вы ожидаете увидеть (эффект стробоскопа).

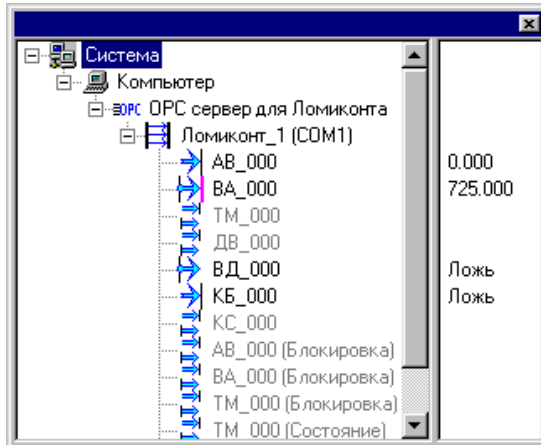
Здесь же можно и редактировать значения переменных. Для этого, после двойного щелчка на значении, в появившемся окне **Ввод значения** введите новое значение переменной. Вид этого окна зависит от типа переменной. Например, если переменная имеет перечислимый тип, то значение выбирается из выпадающего списка. Если необходимо установить значение переменной, которое будет сохраняться при последующих опросах независимо от данных, пришедших по связям, результатов вычислений и прочих обстоятельств, способных изменить значение переменной, то нужно установить флаг **Заблокировать**. Значение заблокированной переменной в дереве отображается в квадратных скобках.



В режиме исполнения в *Дереве Объектов* возможно выделять те или иные элементы, открывать и закрывать объекты и группы. Редактировать дерево объектов в режиме исполнения невозможно.

Дерево системы (RT)

В режиме исполнения *Дерево Системы* принимает вид:



В режиме исполнения окно *Дерева Системы* разделяется на две части вертикальной чертой. В левой части можно видеть само дерево системы в том же виде, в каком оно было в режиме разработки. Справа от разделительной черты отображаются текущие значения OPC переменных.

- ❖ **Важно!** Это поле обновляется периодически, поэтому для данных, изменяющихся быстро и циклически возможны некоторые расхождения с тем, что вы ожидаете увидеть (эффект стробоскопа).

Элементы (OPC серверы, OPC переменные), у которых нет данных отображаются серым цветом. В режиме исполнения в дереве системы возможно выделять те или иные элементы, открывать и закрывать *Компьютеры*, *OPC серверы* и *группы OPC переменных*. Редактировать дерево в режиме исполнения невозможно.






Панель управления

В режиме исполнения *Панель управления* состоит из двух частей: *основной* и *дополнительной*. Описание элементов управления основной панели представлено ниже:

Кнопка Действие

	Завершение сеанса и смена пользователя.
	Показать проект (страницы свойств элементов).
	Показать/Скрыть дерево системы.
	Показать/Скрыть дерево объектов.
	Открыть стартовую мнемосхему.
	Открыть документ родительского объекта (документ того же типа, что открыт в данный момент у дочернего).
	Открыть предыдущий документ(ы).

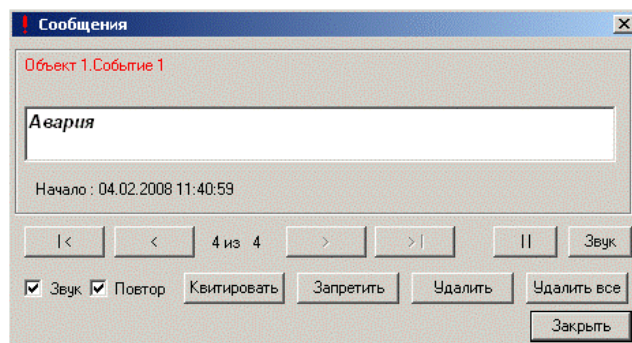
Кнопка Действие

	Открыть следующий документ(ы).
	Перейти в режим разработки.
	Режим работы компьютера и статус связи с другим. Если компьютер работает в режиме основного, кнопка нажата. Если компьютер работает в режиме резервного, - отжата. Кнопка активна только, если имеется связь с другим компьютером. При нажатии кнопки компьютера, работающего в резерве, на основной выдается запрос на переключение.
	Состояние статуса отказа компьютера. В случае, если нет системных отказов позволяет установить или снять пользовательский отказ.
	Показать неквитированные сообщения. Кнопка активна только когда в проекте имеются активные неквитированные сообщения. При нажатии на кнопку открывается Основной журнал компьютера, в котором показаны эти сообщения.

Дополнительная панель содержит кнопки для перехода ко всем документам текущего объекта. Ее вид зависит от того, какие документы имеет объект. В случае отсутствия *Панели управления* нажмите сочетание клавиш **Alt+F10**.

Окно сообщений

Окно Сообщений служит для вывода сообщений на экран в режиме исполнения. С его помощью оператор имеет возможность получать информацию, содержащуюся в сообщении, а также управлять выводом сообщений.












Справа от слова *Сообщения* находится значок, показывающий приоритет текущего сообщения. Приоритетов всего четыре.

- Максимальный - **!!** ;
- Высокий - **!** ;
- Средний - **✓** ;
- Обычный - нет значка.

В текстовом поле выводится текст сообщения. В зависимости от эффектов шрифта сообщения можно различить:

- **Жирный курсив** - Актуальное неквитированное;
- **Жирный** - Актуальное квитированное;
- **Курсив** - Неактуальное неквитированное;
- **Обычный** - Неактуальное квитированное.

Назначения кнопок описаны ниже.

	Показать первое сообщение.
	Показать предыдущее сообщение.
	Показать следующее сообщение.
	Показать последнее сообщение.
	Квитировать.
	Удалить текущее сообщение.
	Удалить все сообщения.
	Закреть окно сообщений
	Нажатие этой кнопки вызовет контекстное меню: Это сообщение; Этой категории; От этого источника; Этого приоритета;



Все сообщения;
Отменить все запреты.
в котором следует выбрать, что именно вы хотите запретить.
Кнопка пауза предназначена для остановки автоматического перехода на следующее актуальное сообщение (если его приоритет не меньше текущего).
Кнопка звук предназначена для перехода к сообщению, от которого запущен звук.
Включить/выключить звук, издаваемый при появлении нового сообщения.
Включить/выключить повтор звука, издаваемого при появлении нового сообщения.

Строка состояния (RT)

В режиме исполнения в *Строке состояния* отображается следующая информация: дата и время возникновения последнего сообщения, имя источника сообщения, и текст этого сообщения.

25.01.2003 14:03:04	Печь. Температура	Минимальное аварийное
---------------------	-------------------	-----------------------

При помощи кнопки в правом углу *Строки состояния* можно просмотреть *Протокол работы* пакета MasterSCADA в режиме исполнения.

Страницы свойств в Режиме исполнения

В режиме исполнения состав и содержимое страниц свойств элементов проекта может отличаться от страниц с теми же названиями, используемых в режиме разработки. В этом разделе справки описаны страницы, характерные только для режима исполнения.

Система

Все страницы свойств Системы в режиме исполнения не отличаются внешним видом от режима разработки, но не доступны для редактирования.

Компьютер

В режиме исполнения Компьютер имеет следующие страницы свойств:

- **Общие** - как в режиме разработки, но поля не доступны для редактирования;
- Резервирование;
- **Режим исполнения** - как в режиме разработки лист *Связь*, но поля не доступны для редактирования;
- Опрос;
- Статистика;
- **Сообщения** - как в режиме разработки;
- **Архив** - как в режиме разработки, но поля не доступны для редактирования;
- **Операторы** - как в режиме разработки - в режиме исполнения существует возможность изменения списка операторов;
- **Смены** - как в режиме разработки, но поля не доступны для редактирования.

Жирным цветом выделены страницы свойств, на которых можно только получить информацию о значении тех или иных параметров, которую вы ввели при разработке проекта. Страница выглядит также, как в режиме разработки. Обычным отмечены страницы, характерные для режима исполнения. Закладка **Резервирование** позволяет управлять и вести наблюдения за работой системы резервирования компьютеров.

Режим работы: Основной Перекл. в режим основного

Статус связи с другим компьютером: Связь с резервным

Статус отказа:

Нет отказа

Установить отказ

Игнорировать отказ

Статистика

Разное

- Восстановление данных
- Запросы
- Количество переданных элементов
 - Количество получаемых архивных переменных: 0
 - Количество созданных элементов
- Период синхронизации: 0
- Получение данных
- Сбор данных

Группа *Режим работы* - режим работы данного компьютера (основной или резервный). Группа *Статус связи с другим компьютером* - есть ли связь с другим компьютером. Группа *Статус отказа* - есть или нет отказ компьютера. Если отказ есть, то в списке указываются его причины:

- Выставлен пользователем;
- Отказ связи с контроллером или OPC;
- Отказ, произошедший в результате других ошибок.

Кнопка *Установить отказ* - позволяет снять отказ компьютера, поставленный ранее пользователем. Если пользовательского отказа нет, то позволяет его поставить. Установить или снять пользовательский отказ можно также кнопкой на панели управления. Кнопка *Игнорировать отказ* - при нажатой кнопке компьютер работает в нормальном режиме (без отказа). Кнопка *Перекл. в режим основного* - действие при нажатии этой кнопки у резервного компьютера аналогично действию при нажатии кнопки *Режим работы компьютера* на панели управления, т.е. на основной компьютер выдается запрос на переключение. Группа *Статистика* - отображаются некоторые параметры, характеризующие работу системы резервирования компьютеров.

Закладка *Опрос* имеет те же поля, что в режиме разработки, но доступные только для чтения, плюс отображает следующие данные о рабочих потоках:

Загруженность рабочих потоков: 0% 100%

Максимальное значение: Сейчас:

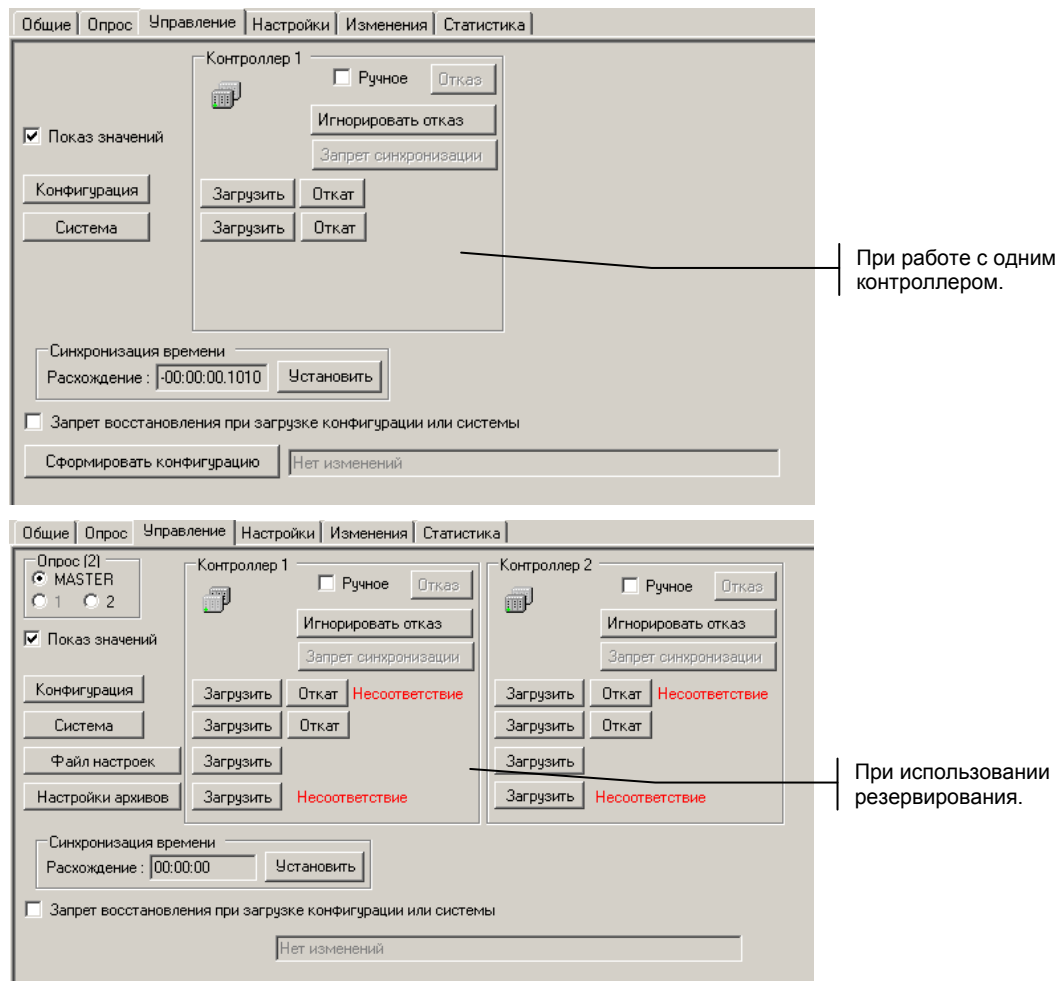
Индикатор *Загруженность рабочих потоков* показывает, на сколько процентов используются рабочие потоки. *Максимальное значение* - максимальное значение рабочих потоков. *Сейчас* - Число рабочих потоков в настоящий момент.

Закладка *Статистика* позволяет наблюдать за распределением ресурсов компьютера по циклам вычислений.

Название	Задано	Среднее	Минимум	Максимум	Приорите...
Система.Компьютер 2	1000	1000	1000	1047	50.000
Котельные.График	1000	1000	1000	1047	50.000

Контроллеры

Вид страницы свойств:



Функции элементов управления описаны на странице ниже:

Элемент	Описание
Группа Конфигурация	При нажатии кнопки <i>Загрузить</i> , конфигурация загружается заново.
Группа Исполнительная система	При нажатии кнопки <i>Загрузить</i> , исполнительная система загружается заново.
Поле Показ значений	Если флаг не стоит, значения в деревьях проекта отображаться не будут.
Группа Файл настроек	Здесь можно загружать файл values.bin, где хранятся настройки контроллера (если включена настройка <i>Хранить настройки контроллера в отдельном файле</i>), смотреть отличаются ли текущие настройки от загруженных. При загрузке этого файла перезагрузка проекта в контроллере не выполняется.
Группа Настройки архивов	Здесь можно загружать файл, где хранятся настройки архивирования, в контроллер, смотреть отличаются ли текущие настройки проекта от загруженных настроек в контроллер. При загрузке этого файла перезагрузка проекта в контроллере не выполняется.
Группа Контроллер	Если стоит флаг <i>Ручное</i> , то при нажатии кнопки можно изменить статус программного отказа контроллера. В случае программного или аппаратного отказа контроллера появляется надпись рядом с картинкой контроллера <i>Отказ</i> . При применении резервирования на странице свойств отображаются две группы <i>Контроллер</i> . Рисунок слева обозначает, в каком режиме находится данный контроллер. Если контроллер основной, то ярким является его изображение, находящееся на переднем плане, если резервный - то на заднем плане. Зеленый огонек означает, что с контроллером установлено соединение. Если с контроллером нет соединения, то горит красный порт на изображении контроллера.
Поле Синхронизация времени	Выдается отклонение системного времени контроллера <i>MASTER</i> . По кнопке <i>Установить</i> время компьютера принудительно записывается в контроллер (даже если отключена автоматическая синхронизация).

Объекты

Страницы свойств объектов:

- **Общие** – аналогичны режиму разработки, но поля не доступны для редактирования;
- **Опрос** - аналогичны режиму разработки, но поля не доступны для редактирования;
- Окна;
- Тренды;
- Рапорты;
- **Расписание** - аналогичны режиму разработки, но поля не доступны для редактирования;
- **Архив** - аналогичны режиму разработки, но поля не доступны для редактирования;
- Паспорт;
- Другие.

Жирным выделены страницы свойств, на которых возможно только получить информацию о значении тех или иных параметров, которую ввели при разработке проекта. Страница выглядит также, как в режиме разработки. *Курсивом* отмечены страницы, характерные для режима исполнения. На закладке **Окна** в режиме исполнения появляются кнопки *Открыть*, *Напечатать*, *Сохранить* и *Открыть сохраненные*. В остальном, страница свойств выглядит как в режиме разработки.

- Кнопка *Открыть* служит для открытия окна документа;
- Кнопка *Напечатать* предназначена для печати мнемосхемы на принтер;
- Кнопка *Сохранить* предназначена для сохранения текущего состояния мнемосхемы в графическом файле, имеющем формат *.jpg*. Имя файла состоит из даты и времени на момент сохранения;
- Кнопка *Открыть сохраненные* - открывает папку, в которой хранятся графические файлы с изображением мнемосхем (создаются при нажатии в режиме исполнения на кнопку *Сохранить*).

Закладка **Тренды** содержит перечень имеющихся у данного объекта трендов и список их переменных. Чтобы открыть один из трендов, выделите его имя в списке и нажмите кнопку *Открыть*. Функции кнопок аналогичны кнопкам из закладки **Окна**.

Вид закладки Паспорт приведен на картинке ниже.

Шаблон:						
Паспорт						Ввод
Группа	Параметр	Тип	Значение	Сравнение	Уставка	
▶ Технологические парам	Расход	Аналоговый		>		
ТО	Время в работе	Аналоговый	983,437	>	807	
ТО	Время в простоте	Аналоговый	575,937	>	456	

Область **Шаблон** в режиме исполнения не доступна для редактирования. Количество строк и столбцов таблицы и их содержание определено в экземпляре паспорта, для редактирования доступны только столбцы *Значение* и *Уставка*, при наличии разрешения на их редактирование. Кнопка *Ввод* подтверждает окончание ввода. В режиме исполнения при изменении значения параметра или уставки выполняется операция сравнения параметра с уставкой, а также сохранение паспорта. Если параметр или уставка являются переменными, то их сравнение с выдачей сообщений происходит и при закрытом паспорте. Для визуальных функциональных блоков в режиме исполнения есть возможность открыть окно паспорта из мнемосхемы. Для этого нужно щелкнуть правой кнопкой мыши по изображению ВФБ в мнемосхеме. При этом открывается окно, назначение и элементы управления которого в точности соответствуют странице свойств *Паспорт* в режиме исполнения. Вид окна показан ниже:

Насос						
Шаблон:						
Паспорт						Ввод
Группа	Параметр	Тип	Значение	Сравнение	Уставка	
▶ Технологич	Расход	Аналоговый		>		
ТО	Время в работе	Аналоговый	150	>	807	
ТО	Время в простоте	Аналоговый	26,968	>	456	

На закладке **Рапорты** доступны действия: *Печать*, *Просмотр* (как перед печатью), *Открыть сохраненные* (аналогично описанной ранее кнопке).

Закладка **Другие** содержит перечень принадлежащих объекту документов. Чтобы применить к документу одну из имеющихся в распоряжении команд нужно выделить его и нажать соответствующую кнопку.

ОРС серверы и Функциональные блоки

Страницы свойств:

- **Общие** - как в режиме разработки, но поля не доступны для редактирования;
- **Опрос** (у Функциональных блоков) - как в режиме разработки, но поля не доступны для редактирования;
- **Опрос** (у ОРС серверов) - как в режиме разработки, но поля не доступны для редактирования;
- **Свойства** - как в режиме разработки, но поля не доступны для редактирования;
- **Данные**.

Жирным выделены страницы свойств, на которых возможно только получить информацию о значении тех или иных параметров, которую ввели при разработке проекта. Страница выглядит также, как в режиме разработки. *Курсивом* отмечены страницы, характерные для режима исполнения.

Вид закладки **Данные** представлен на картинке ниже.

Имя входа/выхода	Данные	Тип данных	Метка времени	Достоверность
[-] Ломиконт_1 (COM1)				
VA_000	15.000	Веществен...	09:25:08	
TM_000	0	Целый	09:24:53	Нет соединения
DV_000	Ложь	Логический	09:24:53	Нет соединения
VD_000	Истина	Логический	09:25:08	
KS_000	Ложь	Логический	09:24:53	Нет соединения
AV_000 (Блокировка)	Ложь	Логический	09:24:53	Нет соединения
VA_000 (Блокировка)	Ложь	Логический	09:24:53	Нет соединения
TM_000 (Блокировка)	Ложь	Логический	09:24:53	Нет соединения
TM_000 (Состояние)	Ложь	Логический	09:24:53	Нет соединения
DV_000 (Блокировка)	Ложь	Логический	09:24:53	Нет соединения

Закладка содержит таблицу с данными:

Имя входа/выхода	Данные	Тип данных	Метка времени	Достоверность
Имена входов/выходов Функционального блока по группам	Текущие значения входов/выходов	Тип данных каждого входа/выхода	Время записи данных в вход/выход	ОРС - признак качества входа/выхода

Переменные

Страницы свойств:

- **Опрос** - как в режиме разработки, но поля не доступны для редактирования;
- **Данные**;
- **Свойства** (ОРС -переменные) - как в режиме разработки, но поля не доступны для редактирования;
- **Поверка**;
- **Паспорт**;

Жирным выделены страницы свойств, на которых возможно только получить информацию о значении тех или иных параметров, которую ввели при разработке проекта. Страница выглядит также, как в режиме разработки. *Курсивом* отмечены страницы, характерные для режима исполнения.

На закладке **Поверка** методикой поверки задается перечень реперных точек, с которыми сравнивается текущее значение переменной.

<input type="button" value="Проверка"/>		<input type="button" value="Печать"/>		Интервал:	Градусы	
				Методика поверки:	0-30-50-70-100%	
				Класс точности:	5%	
	Значение		Измерение		Отклонение	
	%	°	%	°	%	°
1	0	0	67.0	67.0	67.0	67.0
2	30	30	67.0	67.0	37.0	37.0
3	50	50	67.0	67.0	17.0	17.0
4	70	70	67.0	67.0	-3.0	-3.0
5	100	100	67.0	67.0	-33.0	-33.0

При нажатии кнопки *Проверка* все значения в текущей строке запоминаются, а курсор переходит на следующую строку, в которой отображается и сравнивается с реперной точкой текущее значение переменной. Если отклонение в строке превышает по модулю заданный класс точности, то вся строка выделяется красным цветом и жирным шрифтом и остается такой после запоминания. Также возможно ручное перемещение курсора в любую строку таблицы. При этом значения в строках не запоминаются. Кнопка *Печать* - печать таблицы с добавлением пути, имени переменной, класса точности, даты и времени, фамилии текущего оператора и сохранением шрифтового выделения отклонений. Число строк таблицы соответствует заданному для данной методики числу реперных точек. Столбцы:

- *Значение* - заполнен на основании настроечных данных. Здесь указан процент, соответствующий данной реперной точке и вычисленное значение из интервала (в единицах измерения), соответствующее этому проценту;
- *Измерение* - текущее значение переменной в процентах и в единицах измерения;
- *Отклонение* - разность *Измерение* - *Значение* в процентах и единицах измерения.

Порядок вычислений и передачи данных

Функциональные блоки

В этой главе (в отличие от всей остальной документации) термин функциональный блок (ФБ) используется в расширенном значении. Под ФБ, наряду с обычными ФБ и ВФБ, также понимаются все типы переменных объектов. Это обусловлено тем, что с точки зрения порядка вычислений и передачи данных их функциональность можно условно представить в виде стандартных функциональных блоков:

- **Расчет** – ФБ имеет набор входов, соответствующих переменным расчета, и один выход, на который устанавливается результат вычисления формулы;
- **Событие** – аналогично расчету, с той разницей, что выход всегда имеет логический тип. Кроме того, смена значения с *Ложь* на *Истина* инициирует сообщение;
- **Команда** и **Значение** – ФБ имеет один вход и один выход того же типа. Вычисление заключается в передаче текущего значения входа на выход.

Циклы вычислений

Система в режиме исполнения представляет собой набор *циклов вычислений*. В простейшем случае под циклом вычисления понимается объект в дереве объектов проекта со всеми входящими в него переменными, ФБ и ВФБ. Подчиненные объекты входят в тот же цикл вычислений только в том случае, если не имеют собственного цикла вычисления. Каждый цикл вычислений обеспечивает циклическое выполнение с заданным периодом функциональных блоков в той последовательности, которую они имеют в дереве проекта.

Под тактом (итерацией цикла) понимается однократное последовательное вычисление всех входящих в цикл ФБ. При работе цикла некоторые блоки могут вычисляться не на всех тактах (итерациях). Вопрос – *Вычислять ли блок на данном такте?* решается в зависимости от настроек метода опроса данного блока, а также от метода опроса его входов. При этом учитывается время последнего вычисления блока и изменений его входов. Последовательность выполнения такта указана ниже

В начале, происходит считывание данных с внешних входов и распределение их по входам функциональных блоков. Включает в себя получение данных от таких возможных отправителей данных:

- Выходы источников данных, расположенных в другом цикле вычислений;
- Мнемосхема (включая ее выходы, а также данные от контролов, относящихся к визуальным блокам);
- Выходы OPC серверов;
- Другие компьютеры.

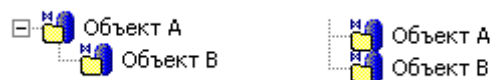
Далее происходит обработка внутренних данных. Заключается в последовательном выполнении функциональных блоков. При вычислении ФБ обновляются данные на его выходах. Если выход связан с входом ФБ в том же цикле вычислений, то данные по связи передаются сразу же. Если для выхода ФБ установлено архивирование, то данные в архив записываются сразу же при изменении значения этого выхода. После этого выполняется передача данных на внешние выходы и распределение их по всем входам, связанным с этими выходами. Этот этап включает в себя передачу данных следующим получателям:

- Входы получателей данных, расположенных в другом цикле вычислений;
- Мнемосхема (включая ее входы, а также данные для контролов, относящихся к визуальным блокам);
- Входы OPC серверов;
- Тренд;
- Другие компьютеры.

Далее возникает пауза до окончания периода опроса.

Каждый цикл вычислений соответствует **Объекту** в дереве проекта. **Объект** может иметь два принципиально различных режима работы в режиме исполнения:

- Объект имеет свой цикл вычислений. В этом случае объект выступает как независимая от родительского объекта сущность.



Если А и В имеют разные циклы вычислений, то в обоих примерах они будут работать параллельно.

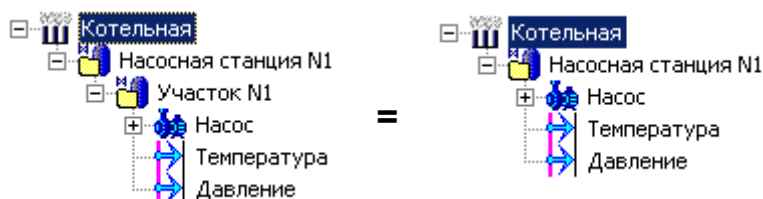
- Объект не имеет своего цикла вычислений. В этом случае объект вычисляется в цикле своего родительского объекта:



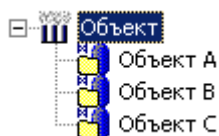
Если В и С имеют общий цикл вычислений, то в обоих примерах они будут работать последовательно.

Порядок вычислений:

- Объекты, не имеющие своих циклов, и функциональные блоки, расположенные на одном уровне иерархии, вычисляются последовательно.
- Вычисление объекта, не имеющего своего цикла, заключается в вычислении его дочерних элементов по предыдущему правилу, как если бы его элементы были просто вставлены на место объекта в списке элементов родительского объекта.



Корневой узел дерева объектов не имеет собственного цикла вычислений, т.е. все его дочерние элементы должны иметь свои циклы. Таким образом, на приведенном ниже рисунке объекты А, В и С всегда будут иметь собственные циклы вычислений.



Если объекты, входящие в один цикл, приписаны разным компьютерам, то на каждом компьютере создается отдельный цикл вычислений, а при переходе в режим исполнения в протокол вносится соответствующее предупреждение.

- ❖ **Важно!** *Асинхронное выполнение различных циклов приводит к повышению общей производительности системы, однако назначать объекту работу в собственном цикле можно только при полной уверенности в том, что это не приведет к потере данных при его взаимосвязях с другими источниками и потребителями данных. Приведем пример - в одном цикле значение дискретной переменной поочередно меняется ($X = NE X$), в другом цикле стоит счетчик, который считает импульсы от этой переменной. Если второй цикл работает в два раза реже первого, то мы можем обнаружить, что счетчик вообще не увеличивается. Это же касается связей между компьютерами, с документами, с OPC-серверами!*

Для **Объектов, имеющих цикл вычислений**, устанавливается минимальный период этого цикла (на закладке *Опрос объекта*). В конце каждого такта определяется время выполнения всего такта. Если оно меньше установленного периода, то в ожидании начала следующего такта делается пауза. Для функциональных блоков производится настройка их методов опроса. Возможные варианты:

- *Всегда*. Блок пересчитывается на каждом такте;
- *Пересчет блока на каждом k-ом такте*. Если $k = 1$, то это эквивалентно случаю 1;
- *Опрос по изменению*. Этот опрос имеет смысл только для входов. При настройке опроса блока *по изменению* заданные значения используются только в качестве настроек по умолчанию для входов блока. В этом случае, если значение на входе изменилось с момента последнего пересчета блока, то блок будет пересчитан в данном такте. Если для входа (или у связанного с ним выхода) задана шкала (диапазон возможного изменения значения) интервал изменения, то мертвую зону (зону нечувствительности) можно указывать в процентах.

Часто встречаются ситуации, когда результаты некоторых вычислений необходимы лишь в определенные промежутки времени. Например, когда формируется рапорт, использующий эти результаты, или открыта мнемосхема, их отображающая.

Чтобы указать, что конкретный элемент должен опрашиваться/вычисляться только при наличии потребителей, в его настройках нужно снять флажок **Включен при отсутствии потребителей**. В режиме исполнения наличие потребителей определяется по следующему правилу - если один из выходов данного функционального блока связан с входом функционального блока, имеющего потребителей, то считается, что у этого ФБ есть потребитель. Условия прекращения опроса при отсутствии потребителей:

- Если у всех выходов ФБ нет потребителей, то ФБ перестает пересчитываться, а на все входы ставится признак отсутствия потребителей;
- Если на входе ФБ установлен признак отсутствия потребителей, то признак передается на связанный выход

Условия возобновления опроса при появлении потребителей:

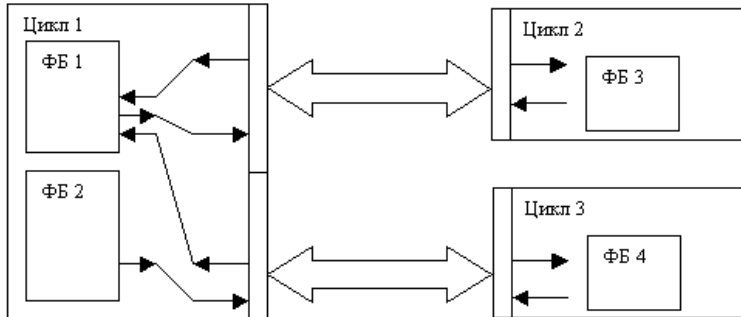
- При открытии документа (мнемосхемы или тренда), где используется такой элемент (переменная или ВФБ), начинается регулярный опрос с заданными настройками;
- При открытии рапорта, используемая в нем переменная однократно опрашивается для получения актуального значения.

Если переменная не обновляется по причине отсутствия потребителей, то ее значение в дереве проекта отображается серым цветом.

- ❖ **Важно!** *Если для переменной задано архивирование, то опрос продолжается независимо от наличия потребителей.*

Внешние связи цикла вычислений

Между независимо функционирующими элементами проекта данные передаются **пакетным (групповым) образом**. Прежде всего это относится к передаче данных между различными циклами вычислений, между циклом вычислений и документами: мнемосхемами (включая их входы, а также контролы, соответствующие визуальным блокам) и трендами, между компьютерами. Связи между циклами вычислений изображены на рисунке ниже.

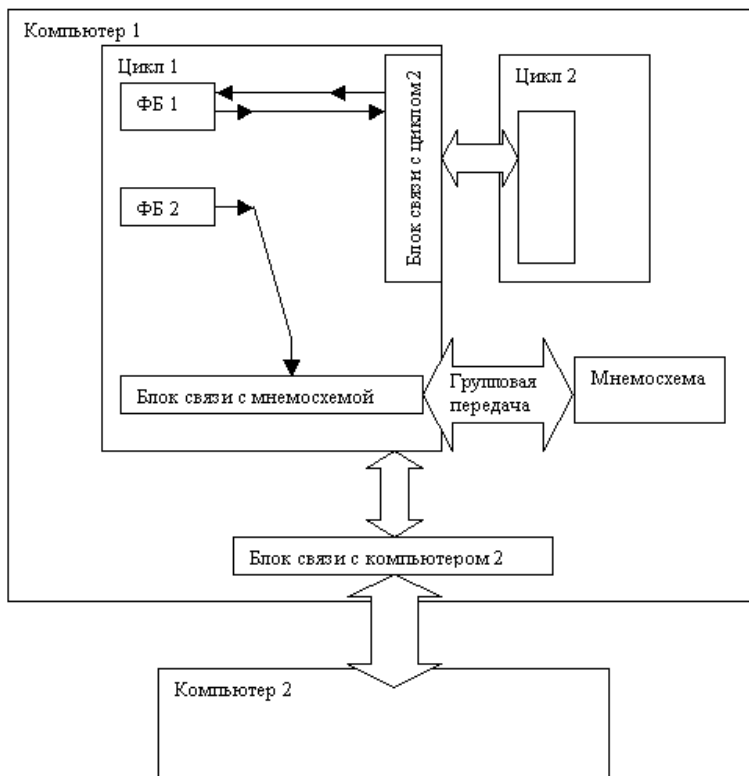


Стрелками вида \longleftrightarrow обозначены групповые (пакетные) передачи данных. Обычными стрелками обозначены связи между одиночными входами/выходами

Порядок пакетной передачи данных:

- Сторона-отправитель в течение заданного периода опроса накапливает измененные данные. Период опроса документов совпадает с периодом опроса объекта, к которому они относятся;
- Все накопленные данные собираются в пакет и отправляются получателю через специальный компонент – *блок связи*;
- В начале такта цикла вычисления получателя, данные пересылаются по связям внутри получателя.

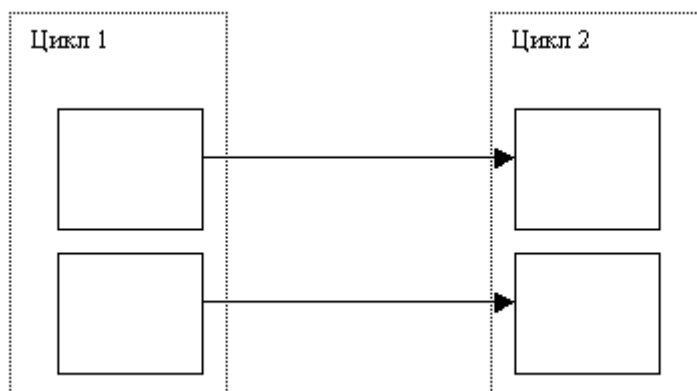
Связи с документами и с другими компьютерами представлены на рисунке ниже.



Для того чтобы обеспечить предсказуемость передачи данных из одного цикла вычислений в другой, реализован механизм **синхронизации передаваемых данных**. Данные по параллельным связям передаются одновременно. Это обеспечивается за счет того, что передача

данных между компонентами системы происходит групповым образом: за один вызов передаются значения всех изменившихся выходов.

- **Пример:** пусть два выхода из цикла 1, связаны с двумя входами в цикле 2. Гарантируется, что данные по двум связям из цикла 1 появятся на входах блоков цикла 2 в одном такте.



Внешние связи исполнительской системы

Циклы вычислений могут находиться на разных компьютерах. Каждый цикл состоит из функциональных блоков, работающих на одном компьютере. Для того, чтобы минимизировать сетевой трафик, все взаимодействие с определенным компьютером производится пакетным образом с заданным в настройках опроса компьютера периодом. Следует отметить, что источники данных могут находиться в разных циклах компьютера-отправителя, аналогично, получатели данных также могут находиться в разных циклах компьютера-получателя. Несмотря на это, при правильном выборе периодов опроса данные от разных циклов могут быть переданы в одном пакете на удаленный компьютер. Если требуется минимизировать загруженность сети, то период опроса компьютера лучше выбирать больше внутренних периодов опроса. Однако в этом случае возможна потеря промежуточных данных, вычисляемых чаще периода отправки их значений на другой компьютер.

Каждому **ОПС серверу** сопоставлен свой цикл вычислений. Таким образом, достигается синхронизация при распределении данных, полученных от ОПС сервера. Работа такого цикла вычислений заключается в циклическом выполнении следующих этапов:

- *Первый этап.* Проверка состояния ОПС сервера. Если соединения нарушилось, то по определенному алгоритму проходит процесс восстановления связи с ОПС сервером;
- *Второй этап.* Если с момента последнего полного чтения данных из сервера истек определенный промежуток времени (период полного чтения), то выполняется чтение (синхронное или асинхронное – в зависимости от настройки) значений всех выбранных переменных ОПС сервера, доступных в режиме чтения. Прочитанные значения (если чтение асинхронное, то это действие происходит в тот момент, когда сервер обработает запрос на чтение) сразу передаются по внешним связям этого цикла вычислений. Полное чтение данных используется с целью обеспечить обновление отметок времени опрашиваемых переменных в случае, если они не изменялись. Если за длительный период времени не появились данные ни от одной переменной сервера, а все они опрашиваются по изменению, этот метод позволяет убедиться в том, что сервер еще функционирует. Также перед полным чтением данных будет производиться попытка переподключения к переменным, с которыми по каким-либо причинам обнаружился обрыв связи;
- *Третий этап.* Если на входы ОПС сервера пришли данные, то они асинхронным образом отсылаются самому серверу. Эта операция, в зависимости от настроек передачи данных ОПС серверу, либо производится всегда при изменении данных, либо все накопленные изменения отсылаются с определенной периодичностью.

Также может быть установлен периодический опрос изменения переменных ОПС сервера. В этом случае, в момент прихода данных от ОПС сервера эти данные сразу же передаются по внешним связям цикла вычислений (аналогично второму этапу).

Восстановление состояния ФБ при рестарте

В MasterSCADA реализована возможность восстановления данных переменных, а также состояния **Функциональных блоков** при последующих рестартах системы. Для того, чтобы данные элемента проекта были восстановлены, флаг *Восстановление при рестарте* на его странице свойств *Опрос* (для переменных-входов и Функциональных блоков) и *Опрос выхода* (для переменных-выходов), должен быть установлен.

Механизм восстановления реализован следующим образом: данные всех элементов проекта, которые требуется восстанавливать периодически, а также при корректном выходе из режима исполнения записываются в специальный служебный файл, и при последующих рестартах из него восстанавливаются.

Состояние *Команд* сохраняется по изменению. Период сохранения настраивается на странице свойств системы *Опрос* в группе *Горячий рестарт*, и может быть переопределен для каждого конкретного компьютера. Настройка *Период актуальности рестарта*, задаваемая на той же странице свойств *Опрос*, определяет интервал времени, отсчитываемый с момента последнего сохранения, в течение которого сохраненные данные считаются актуальными. Т.е. если с момента последнего сохранения данных прошло времени больше, чем указано, то они восстановлены не будут. Очевидно, что сохранение состояния функциональных блоков имеет смысл только для тех ФБ, которые *обладают памятью*, т.е. в процессе работы накапливают необходимые им данные. На сегодняшний день поддержка сохранения состояния реализована в функциональных блоках приведенных по категориям ниже.

Категория *Исполнительные механизмы*:

- Насос мнемонический;
- Клапан переключения потока;
- 2-позиционный механизм с выбором изображения;
- Задвижка;
- Воздуходувка;
- Насос;
- Задвижка мнемоническая;
- Динамический ИМ;
- КПП мнемонический.

Категория *Управление*:

- ПИД-регулятор аналоговый.

Категория *Обработка сигналов*:

- Счетчик пробега;
- Триггер.

Резервирование компьютеров

Резервированный комплект - две одинаковые программы MasterSCADA, установленные на два компьютера, каждая из которых имеет свой ключ. Один из компьютеров является *MASTER* (основным), другой – *SLAVE* (резервным). Выбор статуса компьютера происходит в диалоговом окне при запуске проекта. При этом компьютер *SLAVE* может работать в режиме основного (производить опрос устройств и выполнять вычисление), а *MASTER* в режиме резервного. Отличие компьютера *MASTER* от компьютера *SLAVE* заключается в следующем – при запуске *MASTER* или снятии признака *Отказ* на *MASTER*, происходит переключение управления с *SLAVE* на *MASTER* (с предварительной выдачей сообщения и его подтверждении на *SLAVE*).

Компьютер, работающий в режиме **основного**, выполняет следующие задачи:

- Выполняет проект в полнофункциональном режиме;
- Собирает данные и отправляет по запросу в компьютер, работающий в режиме резервного. Эти данные включают в себя текущие значения всех параметров проекта и состояния всех ФБ, изменившиеся с момента последнего запроса, наборы значений всех архивируемых переменных, накопленные с последнего запроса, набор всех сообщений, произошедших с момента последнего запроса, а также вайлы, измененные

или созданные функциональными блоками (такие, как *Сохранение в файле*, *Данные для рапорта* и т.д.) и сохраненные копии документов (снимки мнемосхем, трендов, сохраненные рапорты);

- Передает в резервный компьютер системное время;
- Обрабатывает запрос от резервного компьютера на переключение;
- Формирует признак «Отказ» (запрет на переключение) при ошибках связи с контроллерами и OPC серверами, а также при ошибках другого рода.

Компьютер, работающий в режиме **резервного**, выполняет следующие задачи:

- Проект с точки зрения пользователя выглядит так же, как в режиме исполнения, но опрос устройств не производится и объекты не выполняются;
- Пользователь имеет возможность производить процедуру *Logon/Logoff* (но в журнал об этом запись идти не будет), просматривать документы, выполнить запрос на переключение (на основном будет выполнен пункт обработки запроса от резервного ПК), установить и снять признак *Отказ*;
- Периодически у основного компьютера запрашиваются накопленные на нем данные. Это обеспечивает идентичность архивных и текущих данных на обоих компьютерах.

Требования к проектам на компьютерах. Для корректной работы на обоих компьютерах должны запускаться идентичные проекты. Для этого необходимо создать проект (или произвести изменения) на одном из компьютеров и затем выполнить экспорт этого проекта на другой компьютер в папку с проектами MasterSCADA (или в любую другую папку компьютера, открытую для записи, а потом скопировать его в папку с проектами). Если экспорт проекта выполняется повторно, то нужно заменить старый проект. При этом в окне MasterSCADA старый проект должен быть закрыт. В случае замены уже существующего проекта архивы старого проекта не удаляются. Экспорт проектов можно производить как в режиме редактирования (меню *Проект/Экспортировать*), так и в режиме исполнения (меню *Сервис/Экспортировать*).

Система резервирования компьютеров в проекте выглядит как один компьютер, которому задают соответствующие настройки на закладках *Общие* и *Режим исполнения* лист *Связь*. Если в системе используется резервирование контроллеров, то в настройках *порта Ethernet* для связи с верхним уровнем необходимо указывать IP-адреса обоих компьютеров.

Наблюдать и управлять работой резервированного комплекса компьютеров в режиме исполнения можно со страницы свойств *Компьютера Резервирование* и при помощи кнопок на *панели управления*.

Отказ компьютера:

- Если **компьютер работает в режиме основного**, то происходит переключение режимов работы (если другой компьютер работает в нормальном режиме). При этом, если отказ выставляет пользователь, то переключение происходит немедленно. При других источниках отказа переключение происходит через некоторое время. После переключения в режим резервного компьютер в случае отказа другого минимум 2 минуты не может перейти в основной режим;
- Если **компьютер работает в режиме резервного**, то ему выставляется запрет на переключение в режим основного.

Снятие отказа компьютера:

- Если это основной компьютер (*MASTER, работающий в режиме резервного по причине отказа*), то на другой компьютер передается запрос на переключение;
- Если это резервный компьютер (*SLAVE*), то с него снимается запрет на переключение в режим основного.

Функциональные возможности диалоговых окон.

Окно-запрос на переключение. Выдается на основном компьютере:

- Если оператор основного компьютера подтверждает запрос, то происходит переключение режимов;
- Если оператор основного компьютера отказывает в переключении, то на резервный компьютер выдается соответствующее сообщение об отказе;
- Если в течение минуты не будет ответа оператора основного компьютера на запрос, то через 60 сек. переключение произойдет автоматически. Окно-запрос после переключения закрывается.

Если на резервный компьютер поступили еще не все архивные данные, то на него выдается соответствующее окно-запрос о подтверждении переключения с предупреждением о потери архивов в случае переключения. Если оператор подтверждает запрос, то на основном компьютере открывается окно-запрос на переключение.

Другие функциональности:

- Если за один цикл синхронизации на резервный компьютер не все накопленные архивные значения будут считаны (то есть количество накопленных архивных значений больше количества передаваемых архивных значений), то их чтение продолжится при следующей синхронизации, и так до тех пор, пока не будут прочитаны все архивы;
- Мгновенные значения передаются при каждом цикле синхронизации независимо от того, все ли архивные значения получены;
- Перезапуск режима исполнения на основном компьютере. Статус основного ему не возвращается до тех пор, пока с другого компьютера не будут перекачены все архивы. После того, как все архивы получены, выдается запрос на переключение режимов;
- Если происходит останов основного компьютера, то управление сразу переходит на другой (даже если тому выставлен отказ);
- Синхронизируются сохраненные документы объектов, такие как: мнемосхемы, тренды и т.д.

Изменение списка операторов в режиме исполнения

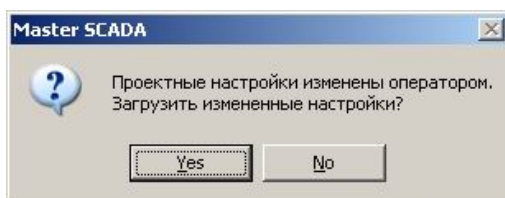
Иногда в процессе работы системы возникает необходимость корректировки списка операторов, а также распределения их по сменам. В MasterSCADA существует возможность выполнения таких действий не останавливая режим исполнения. Если оператор имеет права доступа на действие *Изменение списка операторов* из категории *Проект* (страница свойств системы *Операторы*), то ему в режиме исполнения разрешены следующие действия:

- Добавление/Удаление операторов;
- Назначение паролей операторов;
- Перераспределение операторов по сменам.

В режиме исполнения все вышеприведенные действия производятся с помощью тех же страниц свойств, что и в режиме разработки:

- Корректировка списка операторов, и изменение их паролей - страница свойств *Компьютера Операторы*;
- Перераспределение операторов по сменам - страница свойств *Компьютера Смены*.

Изменения, созданные в режиме исполнения по умолчанию в проекте не сохраняются, но восстанавливаются при последующих запусках системы. Для того, чтобы сохранить эти изменения в проекте, либо наоборот отменить их, необходимо в режиме разработки открыть одну из страниц свойств (*Операторы* или *Смены*) того компьютера, настройки которого менялись. При этом, появится следующее диалоговое окно:



При утвердительном ответе, все изменения, внесенные в список операторов в режиме исполнения, сохраняются в проекте. Иначе, проектные настройки не меняются, при этом созданные изменения сбрасываются.

Исполнительная система контроллера

Ограничение на использование ФБ и формул в контроллере

Прежде чем использовать функциональные блоки из палитры MasterSCADA, убедитесь в том, что они реализованы в исполнительной системе контроллера. В контроллере не реализованы блоки, работающие со строками, с временем, с буферами данных, с архивом, с файлами, с признаками качества и нарушения границ.

Список функциональных блоков, реализованных в контроллере:

Категория	Наименование ФБ
Исполнительные механизмы	Насос мнемонический
	2-поз. мех. в выб. изобр.
	Задвижка
	Воздуходувка
	Насос
	Задвижка мнемоническая
	Динамический ИМ
	КПП мнемонический
	Клапан переключения потока
	Регулирующий клапан с импульсным управлением
ФБ Протокол	
Управление	2-поз. регулятор
	Циклограмма Не формируется выход <i>Имя шага</i>
	ПИД-регулятор
Алгоритмы VNS	Деление с доп. выходами
	Счетчик дискретных импульсов
	X:=Y if A=B
	Контроль равенства по значению
	Контроль диапазона
	Звено запаздывания Необходимо указывать максимальное число хранимых значений
	Максимум из нескольких знач
	Минимум из нескольких знач
	Ограничитель
	Счетчик пробега
	Выдержка времени
	Импульс по времени
	Переключатель 1 к N
	Переключатель N к 1
	Программный задатчик
	Распаковка аналогового 32_бит значения
Упаковка аналогового 32_бит значения	
Контроль параметра	
Пила	

Категория	Наименование ФБ
Обработка сигналов	Пульсатор
	Шум
	Контроль верхней границы
	Контроль нижней границы
	Дискретное среднее
	Дискретный интеграл
	Приводимый дискретный интеграл
	Приводимый скользящий интеграл
	Скользящее среднее
	Скользящий интеграл
	Задний фронт
	Передний фронт
	Триггер
Служебные	Фильтр первого порядка
	Дифференциал
	Антидребезг
	Ограничитель скорости
	Повторитель
	Текущее время

Формулы для вычисления *Расчета* и *События* работают в полном объеме, за исключением функций обращения к архивам, функций работы со временем и строками, функций работы с признаками качества и с признаками нарушения границ.

Подготовка контроллера к работе с MasterSCADA

Подготовка контроллера ADAM-5510

Для загрузки исполнительной системы в контроллер ADAM-5510 нужно проделать следующие операции:

- Установите с диска, прилагаемого к контроллеру, программу *Advantech\ADAM-5510 Series\ADAM-5510SeriesUtility.exe*;
- С ее помощью загрузите на диск D контроллера файлы *ADAM55.EXE* и *autorun.bat*. Как это сделать описано в руководстве к этой программе. Файлы находятся в папке *C:\Program Files\InSAT\MasterSCADA\Config\MasterPLC\ADAM5510*.

Подготовка контроллера Ломиконт

- Установите на компьютер протокол *NetBeui*. За рекомендациями обращайтесь в службу технической поддержки компании ИнСАТ;
- После запуска контроллера нажмите клавишу *ESC*;
- Запустите из командной строки *Norton Commander* (команда *NC*);
- Зайдите в папку *NetMS*;
- Подключите папку компьютера, в которой содержится исполнительная система контроллера (она по умолчанию находится по адресу: *Program Files\InSAT\MasterSCADA\Config\MasterPLC\Ломиконт*). Эта папка должна быть полностью доступна в сети. Используйте команду: *net use*, например, *net use d: \шмя компьютера\Ломиконт*;
- Создайте резервные копии файлов: *autoexec.bat*, *config.sys*. Для этого создайте папку (например, *backup*), скопируйте в эту папку указанные файлы;

- Скопировать содержимое папки Program Files\InSAT\MasterSCADA\Config\MasterPLC\Ломиконт в контроллер;
- Перезагрузите контроллер.

Подготовка контроллера серии I7000\8000

Для каждого из поддерживаемых типов контроллеров используется отдельная исполнительная система (системное программное обеспечение). Исполнительная система включает в себя следующие файлы:

- *autoexec.bat* - пакетный файл для загрузки исполнительной системы;
- *I<тип_контроллера>.exe* - исполняемый файл системы. Например для контроллера 7188E: *I7188E.exe*;
- *7188xw.ini* - конфигурационный файл для программы связи с контроллером.

Данный файл содержит следующие строки:

```
C3 B115200 P0 D8 S1
F
Хautoexec.bat Xi8000.exe
w25
```

С-номер порта

В - скорость обмена

Р-четность

Д-Размер байта

С-стоп бит

Хautoexec.bat Xi8000.exe -устанавливает файлы для автозагрузки

w25 - устанавливает разрешение дисплея в колонках (по умолчанию 25 - макс 50)

7188xw.exe. Файл не загружается в контроллер, служит для настройки параметров связи с контроллером (№ порта, скорость передачи данных и т.п.). Перед началом работы с новым контроллером необходимо загрузить комплект перечисленных файлов в контроллер. Этот комплект находится в папке Program Files\InSAT\MasterSCADA\Config\MasterPLC во вложенной папке с названием контроллера. В эту же папку необходимо загрузить программу связи с контроллером *7188xw.exe* (<http://www.icpdas.com/download/7188xw7188x.htm>).

Для загрузки исполнительной системы в контроллер нужно сделать следующие операции:

- Контакт *INIT* контроллера необходимо соединить с контактом *GND* (для *I-7000*) или с контактом *INIT COM* (для *I-8000*);
- Включите питание контроллера;
- Запустить программу связи с контроллером;
- В появившемся окне необходимо набрать команду *del*, которая предложит удалить уже существующие файлы в контроллере;
- После удаления файлов в контроллере, нажмите *ALT+F9*. После этого начнется загрузка исполнительной системы в контроллер;
- Выключите питание контроллера;
- Контакт *INIT* контроллера разъедините с контактом *GND* (для *I-7000*)или с контактом *INIT COM* (для *I-8000*);
- Для выставления адресов контроллеров необходимо в файле *autoexec.bat*, загружаемом в контроллер, прописать строку вида;
- *I7188XA.exe /an*, где *n* - адрес данного контроллера.

Подготовка контроллера ТКМ410

Для работы с контроллером ТКМ410 нужно сделать следующие операции:

- Установите соединение между компьютером и контроллером при помощи соединительного кабеля. У контроллера используйте *COM3*;
- Запустите на компьютере стандартную программу *Windows – HyperTerminal*;
- В появившемся окне введите имя соединения. Если окно не появилось, нужно выбрать в меню *File* пункт *New Connection*;
- В следующем окне укажите название *COM*-порта компьютера, к которому подсоединен контроллер;

- Укажите параметры связи с контроллером:
- скорость - 115 200;
- количество бит – 8;
- проверка на четность – нет;
- стоповые биты - 1.

Начальные настройки задаются разработчиком контроллеров и указываются в инструкции по эксплуатации.

- Включите контроллер при нажатой кнопке *КОНФИГ*. В терминале появится меню конфигурирования;
- Нажмите цифру 2 и введите IP-адрес и через двоеточие маску подсети контроллера;
- Перезагрузите контроллер. Он готов к работе с MasterSCADA.

При поставке контроллера *TKM410* вместе со SCADA-системой MasterSCADA в него уже загружена исполнительная система MasterPLC.

Чтобы включить вывод отладочных сообщений на HyperTerminal при работе с MasterPLC, сделайте следующее:

- Откройте окно настройки соединения;
- Выберите в строке *Connect Using* протокол *TCP/IP*;
- В строке *Host Address* укажите IP-адрес контроллера;
- Настройка *Port Number* должен быть по умолчанию 23.

Чтобы записывать отладочные сообщения в файл необходимо:

- В главном меню терминала выбрать *Transfer/Capture Text*;
- В открывшемся окне указать имя файла;
- Нажать *Start*.

Подготовка контроллера *WinCon-8000\WinPac-8000*

Для загрузки исполнительной системы в контроллер WinCon-8000 нужно проделать следующие операции:

- Установите параметры локальной сети контроллера: ip-адрес, маску подсети;
- Из меню Programs запустите WinCon(WinPac) Utility, проведите операцию Save and reboot;
- Подключитесь *FTP-клиентом* к контроллеру (например, можно использовать *Total Commander*);
- Загрузите в контроллер файлы *IWINCON8.EXE* и *MplcStarter.exe*. Эти файл находятся в папке *Program Files\InSAT\MasterSCADA\Config\MasterPLC* во вложенной папке с названием контроллера;
- Опять запустите *WinCon(WinPac) Utility*, перейдите на закладку *AUTO-EXECUTE*, в поле *Programs1* пропишите место нахождения файла исполнительной системы: *\Compact Flash\Temp\FTP\MplcStarter.exe*;
- Выполните действие *Save and reboot*.

Настройка работы с модемом. На данном контроллере есть проблема с передачей командной строки в приложение. Поэтому параметры строки нужно записать в отдельный файл *icfg.txt* и положить рядом с *iwincon8.exe*. В автозапуск вставьте *MplcStarter.exe* без параметров.

Возможные параметры:

- */m4* - указывает что модем подключен к порту *COM4*;
- */b115200* - задает скорость работы с модемом (если не задано, используется 9600);
- */iATE0Q0V1X4&C1&D2&S0S0=0+IFC=0,0;+CBST=71,0,1* - переопределение строки инициализации. По умолчанию строка подходит для модемов *Siemens*. В этом случае строка приведена для модемов *MOXA*.

Подготовка контроллеров *МФК/TKM*

Для каждого из поддерживаемых типов контроллеров используется отдельная исполнительная система (системное программное обеспечение).

Исполнительная система включает в себя следующие файлы:

- *autoexec.bat*, *config.sys* - файлы для автозагрузки системы.

Также директория *MasterPLC* содержит следующие файлы исполнительной системы:

- *start.bat* - пакетный файл для загрузки исполнительной системы;
- *I<тип_контроллера>.exe* - исполняемый файл системы. Например для контроллера МФК: *IMFC.exe*;
- *tcp.cfg* - файл с IP-адресом контроллера (только для контроллеров, поддерживающих Ethernet);
- Пакетный драйвер используемой платы *Ethernet* (для контроллеров, поддерживающих *Ethernet*). Он находится на диске, входящем в комплект поставки процессорного модуля, если *Ethernet* встроенный, либо на диске, входящем в комплект сетевой карты;
- Служебные файлы исполнительной системы (*alt_boot.exe*, *reset.com*).

Перед началом работы с новым контроллером необходимо загрузить комплект перечисленных файлов в контроллер. Этот комплект находится в папке Program Files\InSAT\MasterSCADA\Config\MasterPLC во вложенной папке с названием контроллера.

Для загрузки исполнительной системы в контроллер МФК нужно проделать следующие операции:

- Отредактируйте файл *tcp.cfg*: укажите IP-адрес и маску сети;
- Установите соединение между компьютером и контроллером при помощи соединительного кабеля;
- Запустите стандартную программу *Windows – HyperTerminal*;
- В появившемся окне введите имя соединения;
- В следующем окне укажите название *COM-порта*, к которому подсоединен контроллер;
- Укажите параметры связи с контроллером. Например, для контроллера МФК 4823: скорость - 9600, количество бит - 8, проверка на четность - нет, стоповые биты - 1. (Начальные настройки задаются разработчиком контроллеров и указываются в инструкции по эксплуатации);
- Включите контроллер, в течении минуты появится командная строка;
- Создайте директорию *ipc*. Используйте команду *md*;
- Создайте резервные копии файлов: *autoexec.bat*, *config.sys*. Для этого создайте папку (например, *backup*), скопируйте в эту папку указанные файлы (используйте команду *copy*);
- Перешлите файлы в контроллер. Для пересылки файлов в контроллер МФК 5066 используйте команду *Transfer.exe*, в контроллер МФК 4823 и ТКМ52 используйте командный файл *trans.bat*;
- После ввода данной команды будет ожидаться загрузка файла. Загрузка файла запускается при запуске команды *Send File*, находящейся в закладке *Transfer*;
- После появления окна, необходимо указать протокол *Xmodem* и выбрать файл, который необходимо загрузить в контроллер;
- В появившемся окне будет отображаться ход загрузки файла;
- Перезагрузите контроллер.

В следующий раз для соединения с контроллером используйте следующие параметры связи: скорость соединения 19200, количество бит - 8, проверка на четность - нет, стоповые биты - 1.

Удаление файлов в контроллере

Удаление исполнительной системы контроллера

Для удаления исполнительной системы контроллера необходимо установить соединение, используя программу *HyperTerminal*. Настройки соединения должны быть следующими: скорость соединения 19200, количество бит - 8, проверка на четность - нет, стоповые биты - 1. После этого включите контроллер. В процессе загрузки появится сообщение: *Если за данное время после вывода этого текста не будет нажата клавиша пробел, контроллер загрузи-*

зится с минимальным набором программ и перенаправление консольного ввода/вывода в СОМ1. После появления данного сообщения необходимо нажать клавишу пробел.

Восстановите первоначальные (заводские) файлы *autoexec.bat*, *config.sys* из созданной директории (нами была предложена папка *Backup*). После этого удалите директорию *ipc*. Используйте команду *rd*. После указанных действий перезагрузите контроллер.

Для удаления исполнительной системы из контроллера **Ломиконт**, восстановите файлы из папки *backup*.

Для удаления исполнительной системы из контроллера **WinCon-8000** выполните следующие действия:

- Запустите *WinCon Utility*, перейдите на закладку *AUTO-EXECUTE*, очистите поле *Programs1*;
- Подключитесь *FTP*-клиентом к контроллеру (например, можно использовать *Total Commander*);
- Удалите в контроллере файл *IWINCON8.EXE*.

Для удаления файлов в контроллере **I7000V8000** необходимо выполнить следующие действия:

- Контакт *INIT* контроллера необходимо соединить с контактом *GND* (для I-7000) или *INIT COM* (для I-8000);
- Включите питание контроллера;
- Запустить программу связи с контроллером;
- В появившемся окне необходимо набрать команду *del*, которая предложит удалить уже существующие файлы в контроллере;
- После удаления файлов, выключите питание контроллера;
- Контакт *INIT* контроллера разъедините с контактом *GND* (для I-7000) или *INIT COM* (для I-8000).

Автоматическое обновление исполнительной системы контроллеров

Обновление исполнительной системы может происходить при помощи MasterSCADA. Автоматическая загрузка исполнительной системы в контроллер происходит в одном из трех случаев:

- При переходе в режим исполнения;
- При восстановлении связи с контроллером. Производится только в том случае, если во время работы в режиме исполнения загрузка исполнительной системы не произошла;
- Действия пользователя со страницы свойств *Управление* контроллера.

Загрузка исполнительной системы в контроллер состоит из нескольких этапов:

- Происходит сравнение версии уже загруженной исполнительной системы контроллера с версией, которая хранится в MasterSCADA;
- В случае, если версии различны, происходит запись новой исполнительной системы в контроллер. В случае, если версии одинаковы, то загрузка далее не идет, за исключением случая, когда загрузка инициирована пользователем со страницы свойств *Управление*, с установленным флагом *Принудительно*;
- В случае, если произошла ошибка, то выдается соответствующее сообщение. MasterSCADA повторяет попытку загрузить исполнительную систему (не более трех раз);
- Контроллер перезагружается (в контроллере WinCon происходит перезапуск исполнительной системы, контроллер не перегружается).

Архивирование в контроллере

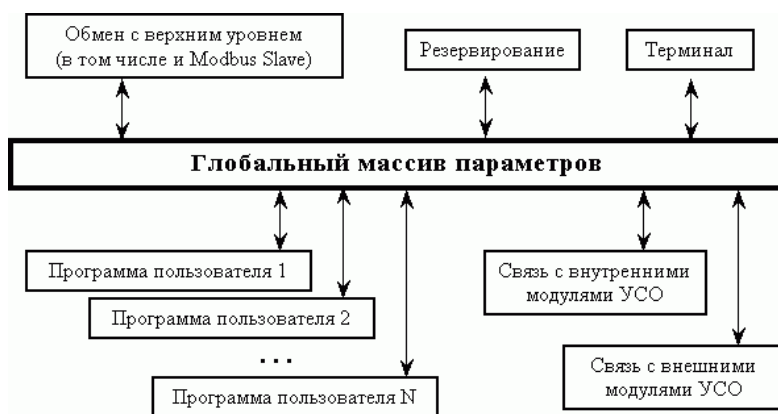
Архивирование данных можно выполнять непосредственно в контроллере. Для этого на странице свойств *Настройки* контроллера должна быть выбрана функция *Выполнять архи-*

вирование в контроллере. На странице свойств *Общие* переменной, данные которой необходимо записывать в *Архив*, должен быть установлен флаг *Архивировать*.

Контроллер в ходе своей работы накапливает данные в архиве. При опросе с верхнего уровня эти данные считываются в архив компьютера через промежутки времени, равные периоду опроса в настройках контроллера. При этом сначала происходит чтение мгновенных значений, а затем чтение архива. Чтение архива оканчивается, когда все накопленные данные получены. Команда чтения архива может быть прервана командой принудительного чтения. Для оптимизации быстродействия часть архива хранится в оперативной памяти контроллера. Если указанный в настройках Контроллера *Максимальный размер архива* не больше размера свободной оперативной памяти, то архив целиком хранится в оперативной памяти и диск не используется. Это имеет смысл, если в контроллере не установлена память *SRAM*. Если архив не умещается целиком в оперативной памяти, то происходит его сброс на диск. Сохранить архив на диск контроллера можно также при помощи переменной *Сохранение данных*. Если функция *Выполнять архивирование в контроллере* не используется, то архивирование тех переменных, для которых установлен флаг *Архивировать*, выполняется только на компьютере. При этом значения в *MasterSCADA*, а соответственно и в архив поступают через промежутки времени, равные *Периоду опроса* в настройках контроллера.

Архитектура системы

Системное программное обеспечение представляет собой один исполняемый файл. Этот файл выполняется под управлением встроенной операционной системы микроконтроллера (*DOS*, *MiniOS7* и др.). При запуске исполнительная система загружает конфигурацию из конфигурационного файла. На основе этой конфигурации ядро исполнительной системы запускает на выполнение пользовательские и системные задачи, реализующие заложенную в конфигурацию при создании функциональность. Конфигурационный файл генерируется инструментальной средой разработки. Архитектура исполнительной системы представлена на рисунке ниже.



Все задачи в исполнительной системе контроллера работают под управлением многозадачного ядра реального времени. Задачи выполняются параллельно, в режиме приоритетной многозадачности. Все задачи пользователя имеют одинаковый приоритет. Если объекты, для которых создавались схемы ФБ, имеют одинаковый период выполнения, то все схемы ФБ выполняются в одной задаче в порядке их расположения в проекте. Если у какого-либо объекта период выполнения отличается, то его схема ФБ выполняется в отдельной задаче.

Загрузка конфигураций

Загрузка конфигурации в контроллер происходит в одном из трех случаев:

- При переходе в режим исполнения;
- При восстановлении связи с контроллером. Производится только в том случае, если во время работы в режиме исполнения загрузка конфигурации не произошла;
- Действия пользователя со страницы свойств *Управление* контроллера.

Загрузка конфигурации в контроллер состоит из нескольких этапов:

- Происходит сравнение уже загруженной конфигурации контроллера с конфигурацией MasterSCADA;
- В случае, если конфигурации различны, происходит запись новой конфигурации в контроллер. В случае, если конфигурации одинаковы, то загрузка далее не идет, и конфигурация считается загруженной, за исключением случая, когда загрузка конфигурации инициирована пользователем, в этом случае, загрузка будет происходить заново;
- В случае, если произошла ошибка, то выдается соответствующее сообщение. MasterSCADA повторяет попытку загрузить конфигурацию (не более трех раз). В случае, если конфигурация так и не была загружена, то опрос контроллера прекращается, при этом проверочная команда продолжает поступать. Как только связь с контроллером восстанавливается, то попытки загрузить конфигурацию повторяются;
- Исполнительная система перезапускается.

Скорость загрузки конфигурации в контроллер. В случае, если запись конфигурации идет по COM-порту, то первоначально MasterSCADA пытается осуществить загрузку на скорости, установленной пользователем на странице свойств *Настройка*. Если в контроллере уже выполняется конфигурация, в которой задана другая скорость для связи с внешним уровнем, то MasterSCADA последовательно попытается осуществить загрузку на одной из скоростей, перечисленных на странице свойств *Настройка*. В случае, если доступная скорость не найдена, выдается сообщение, сигнализирующее об ошибке. Если одна из скоростей подойдет, то после загрузки конфигурации и перезагрузки контроллера связь с ним будет производиться не на подобранной скорости, а на той скорости, которая задана пользователем в режиме разработки.

Резервирование контроллеров

Некоторым технологическим объектам предъявляются повышенные требования к надежности их функционирования. Для повышения надежности работы системы автоматизации таких объектов следует использовать резервирование контроллеров. Для этого на странице свойств *Настройки* контроллера должна быть выбрана опция *Использовать резервирование*. На странице свойств *Настройки порта Ethernet* должен быть установлен флаг *Связь с резервным контроллером*. В данном режиме работы используются два контроллера, в которые загружена одинаковая конфигурация за исключением IP-адресов. Конфигурации загружаются в оба контроллера параллельно, независимо друг от друга. При резервировании идет проверка соответствия конфигураций контроллеров. В случае несоответствия конфигураций механизм резервирования не действует, и контроллеры работают независимо.

В процессе работы резервированного комплекса каждому из контроллеров присваивается один из двух взаимоисключающих статуса:

- *Основной контроллер* выполняет прикладной проект и осуществляет управление объектом автоматизации;
- *Резервный контроллер* выполняет прикладной проект (периодически запрашивает данные у основного контроллера), но не производит управление объектом.

Присвоение, сохранение или изменение того или иного статуса каждого из контроллеров производится по результатам анализа состояния программного или аппаратного отказа контроллера. После запуска системы основным становится тот контроллер, загрузка которого произойдет быстрее. Во время работы контроллеров производится контроль их текущего состояния и диагностика неисправностей. Все неисправности контроллеров делятся на отказы и ошибки:

- *Отказы* – это неисправности, при которых контроллер не в состоянии выполнять возложенные на него функции управления;
- *Ошибки* – это неисправности, при которых сохраняется работоспособность контроллера как управляющего устройства.

Отказ и ошибка могут быть сформированы по инициативе оператора или по событию в прикладной программе. Для этого в контроллере имеются специальные служебные параметры. При возникновении отказа основного контроллера формируется команда на переключение управления с основного контроллера на резервный с одновременным изменением статуса обоих контроллеров комплекса. Переключение управления производится безударно (с ми-

нимальным временем переключения) с точки зрения объекта управления и с точки зрения АРМ (специальная обработка сетевых переключений в режиме исполнения проекта MasterSCADA).

При возникновении отказа резервного контроллера формируется команда на запрет переключения управления с основного контроллера на резервный, тогда как основной контроллер продолжает выполнять технологическую задачу в режиме одиночного (без резерва). Для правильной работы резервированного комплекса необходимо тем переменным и ФБ, данные которых должны передаваться от основного контроллера резервному, назначить **Восстановление при рестарте**. Быстрее и проще это сделать следующим образом: назначить *Восстановление при рестарте* всему объекту, который выполняется в контроллере, и далее убрать флаги *Восстановление при рестарте* у тех переменных и ФБ, которые должны выполняться в обоих контроллерах независимо.

Сообщения возникающие при резервировании контроллеров указаны в таблице ниже:

Текст сообщения	Ситуация
(контроллер N) Нет соединения с контроллером	N - 1, 2 Данное сообщение является условным. С одним из контроллеров комплекса нет связи по следующей причине: 1. После включения комплекса неработоспособен один из контроллеров. 2. Контроллер нормально функционирующего комплекса выходит из строя в целом (пропадание питания, отказ платы CPU, срабатывание сторожевого таймера и т.д.). 3. Ситуация отказа связи между контроллером нормально функционирующего комплекса и АРМ. 4. Производится программный перезапуск контроллера. Другой контроллер комплекса функционирует в режиме основного. Сообщение пропадает при восстановлении связи.
Нет соединения с контроллером	Данное сообщение является условным, оно возникает при обрыве связи с обоими контроллерами и пропадает при восстановлении связи. В то время, пока сообщение активно, обмен данными между MasterSCADA и контроллерами не ведется, и всем переменным выставляется признак качества <i>Нет соединения</i> .
Сообщение от контроллера: Произведено переключение с контроллера 1 на контроллер 2. (с контроллера 2 на контроллер 1)	Произведено переключение управления на другой контроллер по следующей причине: 1. В основном контроллере нормально функционирующего комплекса возникает программный отказ. 2. В нормально функционирующем комплексе производится программный перезапуск основного контроллера. 3. В комплексе, работающем в режиме «Отказ основного контроллера», устраняется отказ основного контроллера. 4. В комплексе, работающем в режиме «Отказ обоих контроллеров», устраняется отказ резервного контроллера.

- ❖ **Важно!** Условное сообщение - это сообщение, которое имеет начало и конец. Под нормальным функционированием комплекса понимается ситуация, когда в отсуствии отказов одновременно работают оба контроллера: один – в режиме основного, другой – в режиме резервного. Контролировать работу резервированного комплекса можно на странице свойств контроллера Управление.

Ресурсы контроллера

Для управления системой внутри контроллера применяется набор служебных параметров Ресурсы. Группа переменных **Диагностика**:

- Вход *Отказ* позволяет выставить программный отказ в контроллере или устранить ранее сгенерированный программный отказ;
- Вход *Ошибка* позволяет выставить программную ошибку в контроллере или устранить ранее сгенерированную программную ошибку;
- Выход *Статус отказа* принимает значение *Вкл* при отказе контроллера (программном или аппаратном);
- Выход *Статус ошибки* принимает значение *Вкл* при обнаружении ошибки в контроллере (программной или аппаратной);
- Выход *Флаги состояния* при аппаратном отказе контроллера возвращает битовые значения, указывающие на тип отказа:

Флаг	Тип отказа
0x0100	ошибка встроенных модулей УСО.
0x0200	ошибка модулей УСО, подключенных по последовательному порту.
0x0400	ошибка задачи резервирования.

Группа переменных **Управление**:

- По переднему фронту входа *Перезапуск* осуществляется перезагрузка контроллера.

Группа переменных **Резервирование** (группа становится доступной, если на странице свойств Настройки контроллера задана функция *Использовать резервирование*):

- Выход *Резерв* выдает признак того, что контроллер находится в резерве;
- Выход *Номер* отображает номер контроллера, на котором выполняется программа;
- Выход *Запрет на управление* имеет значение *Вкл*, если оператором выставлен запрет на управление (включен режим блокировки);
- Выход *Период межконтроллерного обмена* возвращает значение периода, с которым резервный контроллер считывает данные с основного.

Группа переменных **Модем** (группа становится доступной, если на странице свойств Настройки контроллера задана функция *Использовать GSM модем*):

- Выход *Сброс модема* позволяет увидеть, находится ли модем в состоянии перезагрузки. Во время программной перезагрузки модема можно выполнить его аппаратный перезапуск;
- По переднему фронту параметра *Инициативный вызов модем*, подключенный к контроллеру, устанавливает связь с верхним уровнем.

Вход **Сохранение данных** - переменная становится доступной, если на странице свойств Настройки контроллера задана функция *Выполнять архивирование в контроллере* или *Использовать горячий рестарт*. По переднему фронту параметра *Сохранение данных* на диск контроллера сохраняются архивные данные и данные, которые требуется восстанавливать при рестарте.

Исполнительная система контроллера

Исполнительная система предназначена для программирования программируемых логических контроллеров (PLC) с открытой архитектурой, прежде всего PC-совместимые. Перенос исполнительной системы на новые типы контроллеров возможен как разработчиками, так и пользователями. Для получения детальной информации о поддерживаемых типах контроллеров обращайтесь к разработчикам. Минимальные требования к оборудованию: процессор Intel 80186/80188, ОЗУ - 256 Кб. Примеры поддерживаемых контроллеров указаны ниже.

I7188 производства ICP-DAS (Тайвань):

- Архитектура процессора – Intel 80186/80188. Типовой процессор - AMD186ES 20/40 МГц. Память - 256 или 512 Кб ОЗУ, 256 или 512 Кб flash;
- Операционная система – MiniOS7 (вариация MS-DOS). Содержит минимальный набор команд для загрузки файлов в режиме терминала и запуска исполняемых файлов. Имеет частичную совместимость с MS-DOS. Для работы с каналами ввода/вывода, терминалом, таймерами, последовательными портами, портом Ethernet, FLASH-дискон и пр. имеются стандартные библиотеки;
- Имеет набор последовательных портов RS232/RS485. Возможен порт Ethernet;
- Модули ввода-вывода - внешние, подключены через RS485.

2. MFC производства Tecop (Россия):

- Архитектура процессора – Intel 80486, 100 МГц. Память - 4 или 8 Мб ОЗУ, 4 или 8 Мб flash;
- Операционная система – MS-DOS 6.22;
- Имеет набор последовательных портов RS232/RS485 и порт Ethernet;
- Модули ввода-вывода - встроенные, в кейтовом конструктиве.

Соединение через GSM модем

В MasterSCADA имеется возможность работы с контроллером через GSM-модем. Перед началом работы с модемом необходимо настроить его конфигурацию. Для этого нужно сделать следующие операции:

- Установите соединение между компьютером и модемом при помощи соединительного кабеля. Включите питание модема;
- Запустите стандартную программу *Windows – HyperTerminal*;
- В появившемся окне введите имя соединения. Если окно не появилось, нужно выбрать в меню *File* пункт *New Connection*;
- В следующем окне укажите название *COM-порта*, к которому подсоединен модем;
- Укажите параметры связи с модемом:
 - скорость – 115200;
 - количество бит – 8;
 - проверка на четность – нет;
 - стоповые биты – 1;
- В командной строке введите команду *ATE0*, чтобы отключить эхо команд на экране в командном режиме. Модем ответит *OK*;
- Далее чтобы вводимые команды отображались в командной строке, нужно зайти в меню *File/Properties* на закладку *Setting*, нажать кнопку *ASCII Setup...* и в открывшемся окне установить флаг *Echo typed characters locally*;
- Введите команду *AT&W0*, чтобы сохранить рабочую конфигурацию в запоминающем устройстве модема. Модем ответит *OK*;
- Конфигурация модема настроена, можно закрыть программу *HyperTerminal* или нажать кнопку *Disconnect*.

В файле контроллера, выполняющем запуск исполнительной системы (в контроллерах I7000 и в I8000 это файл *autoexec.bat*, в других - *start.bat* в директории MPLC), нужно дописать в строке запуска exe-файла ключ */mN*, где *N* - номер COM-порта контроллера, к которому подключен модем.

Для I8000:

- */m1* - означает связь через модем *comport 1*;
- */c1* - означает связь через *comport 1*;
- */b9600* - означает связь со скоростью *9600*.

Модем принимающий вызов автоматически не должен поднимать трубку. На компьютере COM PORT должен поддерживать сигнал *RING INDICATOR (RI)*

Для того, чтобы контроллеру стал доступен модем, подключенный к его COM-порту, необходимо на странице свойств контроллера *Настройки* выбрать функцию *Использовать GSM модем*. Настройка параметров модема, подключенного к контроллеру, производится на странице свойств COM-порта *Настройки*. Настройка параметров модема, подключенного к компьютеру, производится на закладке *Модемы* страницы свойств компьютера *Опрос*.

После установления соединения MasterSCADA посылает контроллеру команды в следующем порядке:

- Команды проверки состояния контроллера;
- Команды загрузки исполнительной системы;
- Команды загрузки конфигурации;
- Команды управления отладкой контроллера;
- Команды записи в контроллер;
- Команды чтения.

Сначала происходит чтение мгновенных значений, затем если в контроллере выполняется архивирование - чтение архивных значений. В контроллер может быть послана команда принудительного чтения. В этом случае команды чтения становятся по приоритету выше команд записи и отладки. Когда все команды посланы, связь разрывается. Если определено *Максимальное время соединения*, то по истечении этого времени связь разрывается в любом случае, даже если контроллеру посланы еще не все команды. После разрыва связи запрос на установление нового соединения посылается через определенный промежуток времени - *Период опроса*. В случае неудачной попытки соединения следующие запросы посылаются через промежутки времени, равные *Минимальному интервалу между дозвонами*. За

ходом выполнения команд можно наблюдать на странице свойств контроллера *Управление*. Во время работы с модемом MasterSCADA выдает соответствующие сообщения. Соединение через модем может устанавливаться по запросу с нижнего уровня. Смотрите параметр контроллера *Инициативный вызов*. При этом имеется возможность установить соединение не через тот модем, который осуществляет опрос контроллера, а через модем, подключенным к другому COM-порту компьютера. COM-порт компьютера должен быть и соединительный кабель должны быть полномодемными (*Full modem*).

Сообщения при загрузке

При работе с контроллером MasterSCADA выдает следующие сообщения. Условное сообщение - это сообщение, которое имеет начало и конец.

Текст сообщения	Ситуация
Загрузка конфигурации в контроллер: инициализация контроллера	Данное сообщение является условным, оно сигнализирует о начале/ конце загрузки конфигурации в контроллер. Загрузка проходит по инициализации MasterSCADA.
Загрузка конфигурации в контроллер: ручная перезагрузка пользователем	Загрузка происходит в результате действий пользователя.
Ошибка загрузки конфигурации в контроллер	Сигнализирует о том, что в момент загрузки конфигурации в контроллер произошла какая-либо ошибка (например, в момент загрузки произошел сбой связи с контроллером). Данное сообщение является условным, оно возникает при обрыве связи с контроллером и пропадает при восстановлении связи. В то время, пока сообщение активно, обмен данными между MasterSCADA и контроллером не ведется, и всем переменным выставляется признак качества <i>Нет соединения</i> .
Нет соединения с контроллером	Сообщение появляется в случае, если команда, обеспечивающая передачу данных в контроллер, не проходит (MasterSCADA повторяет данную команду 3 раза).
Ошибка записи данных в контроллер	

Типы задач исполнительной системы контроллера

В исполнительной системе выполняются следующие задачи:

- *Программа пользователя* – задача выполнения последовательности функциональных блоков, реализующая определенный алгоритм обработки параметров глобального массива. Количество задач зависит от данных в конфигурационном файле;
- *Связь с внутренними модулями устройств связи с объектом (УСО)* – задача функционирует с определенным периодом, и осуществляет чтение/запись физических контекстов ввода/вывода (данных встроенных модулей ввода-вывода контроллера);
- *Связь с внешними модулями УСО* – осуществляет связь с внешними модулями УСО по коммуникационным каналам связи – последовательным портам и *Ethernet*. Количество задач зависит от данных в конфигурационном файле;
- *Обмен с верхним уровнем* – задача обеспечивает связь с верхним уровнем - системой программирования (MasterSCADA или MasterLogic), SCADA-пакетом или OPC-сервером. Эта задача отвечает за загрузку конфигурации в контроллер, отладку программ пользователя, мониторинг состояния исполнительной системы, чтение/запись параметров глобального массива и пр.;
- *Терминал* – задача предоставляет средства для мониторинга состояния исполнительной системы в режиме терминала. Возможен вариант с подключением к контроллеру клавиатуры и монитора (при наличии таковых портов) и вариант удаленного терминала. В случае удаленного терминала задача использует один из последовательных портов (тот, который предусмотрен в контроллере для подобных целей);
- *Резервирование* – осуществляет задачи резервирования;
- *Межконтроллерный обмен* - осуществляет обмен данными с другими контроллерами.

Последние две задачи являются опциональными и присутствуют в комплекте исполнительной системы только в случае их приобретения. На задачи, использующие каналы связи (по-

следовательные порты и *Ethernet*), накладывается ограничение – к каждому последовательному каналу связи может быть привязана только одна коммуникационная задача. К каналу связи типа *Ethernet* можно привязывать несколько задач.

Кроме описанных задач, дополнительно существует еще одна служебная задача – основной поток приложения, из которого происходит инициализация исполнительной системы, разбор конфигурации, и запуск всех остальных задач. После запуска всех задач этот служебный поток переходит в режим ожидания.

Задача **Программа пользователя** предназначена для выполнения программы пользователя. Количество таких задач и конфигурация каждой из них определяется инструментальной средой разработки. Все задачи пользователя имеют одинаковый приоритет. Если объекты, для которых создавались схемы ФБ, имеют одинаковый период выполнения, то все схемы ФБ выполняются в одной задаче в порядке их расположения в проекте. Если у какого-либо объекта период выполнения отличается, то его схема ФБ выполняется в отдельной задаче. Программа пользователя состоит из последовательности функциональных блоков (ФБ). ФБ представляет собой реализацию какого-либо алгоритма обработки параметров. Каждый ФБ имеет кратность выполнения, равную или большую 1. При кратности, равной 1, ФБ выполняется каждый цикл выполнения программы пользователя, при кратности 2 – каждый второй цикл, начиная с первого, и т.д. При кратности, равной нулю, ФБ не выполняется. Каждая задача обработки программы пользователя имеет свой локальный массив параметров. В начале и в конце каждого цикла работы программы пользователя исполнительная система синхронизирует состояние локального массива задачи с глобальным массивом параметров. В процессе выполнения цикла программы пользователя все ФБ внутри этой программы оперируют только параметрами из локального массива задачи.

Задача **Связь с внутренними модулями УСО** обслуживает контексты ввода/вывода модулей, расположенных непосредственно в том же физическом конструктиве, что и модуль центрального процессора, на котором работает исполнительная система. Задача функционирует циклически с заданным периодом и приоритетом.

Задача **Связь с внешними модулями УСО** предназначена для организации распределенного обмена с внешними модулями УСО по коммуникационным каналам связи. В качестве каналов связи могут использоваться последовательные порты *RS232/RS485* и порт *Ethernet*. Количество таких задач и конфигурация каждой из них определяется инструментальной средой разработки и типом УСО, с которыми происходит взаимодействие. Каждая задача функционирует циклически, со своим периодом и приоритетом.

Задача **Обмена с верхним уровнем** обеспечивает всю функциональность, необходимую верхнему уровню для работы с исполнительной системой контроллера: загрузка конфигурации, отладка программ пользователя, мониторинг. Также она предоставляет свой протокол для чтения/записи параметров.

- ❖ **Важно!** Исполнительная система при загрузке анализирует конфигурационный файл, и сначала инициализирует, а затем запускает задачи тех типов, которые поддерживаются в данной реализации системы. Задачи нераспознанного типа игнорируются.

Удаление конфигураций

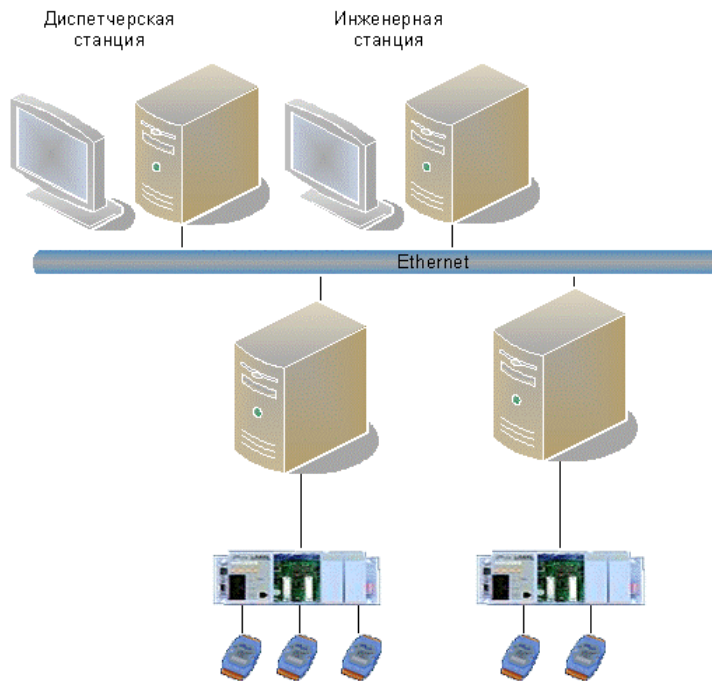
Удаление конфигураций из контроллера происходит следующим образом:

- Если у контроллера есть файловая система вида *Linux*, *Windows CE*, то конфигурацию удаляют посредством удаления файлов *config.bin* или *session.bin*. В случае если существуют и тот, и другой файлы, удалению подлежат оба;
- Если же у контроллера есть своя файловая система *MiniOS*, то конфигурация удаляется с помощью командной строки: *i8000e /d*, где *i8000e* – название исполнительной системы. После названия исполнительной системы ставится пробел.

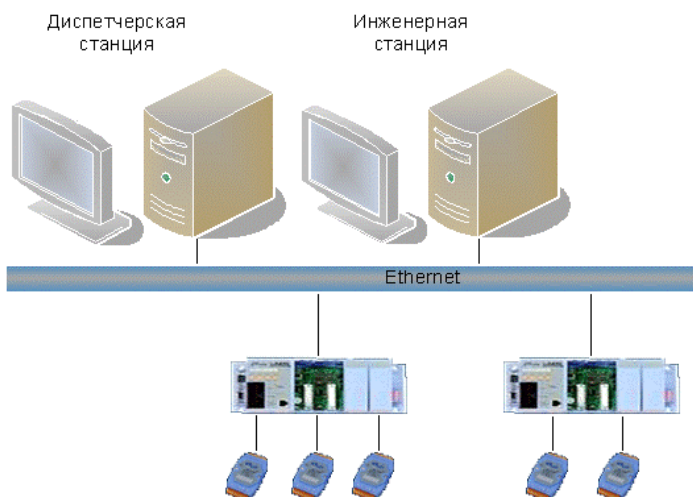
Сетевые настройки MasterSCADA

В основе концепции построения проектов, связанных с сетевым взаимодействием нескольких MasterSCADA положено, что задача автоматизации сбора и обработки информации, выдачи управляющих воздействий, разрабатывается на одной машине как единый проект. А далее, согласно архитектуре системы распределяется между ее узлами: серверами, диспетчерскими и инженерными станциями, контроллерами (если они поддерживают возможность программирования непосредственно из MasterSCADA).

Клиент-серверная архитектура изображена на рисунке ниже:



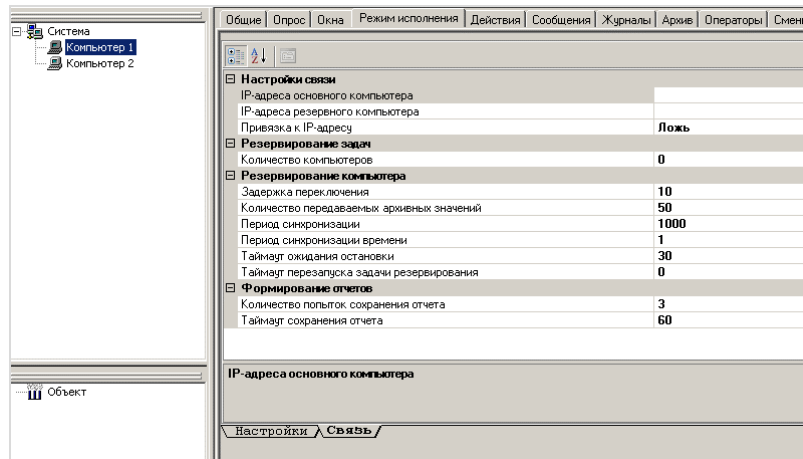
Одноранговая архитектура изображена на рисунке ниже:



Сетевые настройки MasterSCADA:

- В дереве системы выберите компьютер, для которого создаются сетевые настройки;
- Перейдите на закладку *Режим исполнения*;
- Выберите закладку *Связь* (примечание: Для того, чтобы удостовериться в наличии связи между компьютерами, воспользуйтесь командой *ping*, входящей в набор вспомогательных утилит *Windows*);
- В поле *IP-адреса основного компьютера* введите сетевой адрес, назначенный Вашим системным администратором;

- Поле *IP-адреса резервного компьютера* заполняется только при наличии резервированной системы управления.

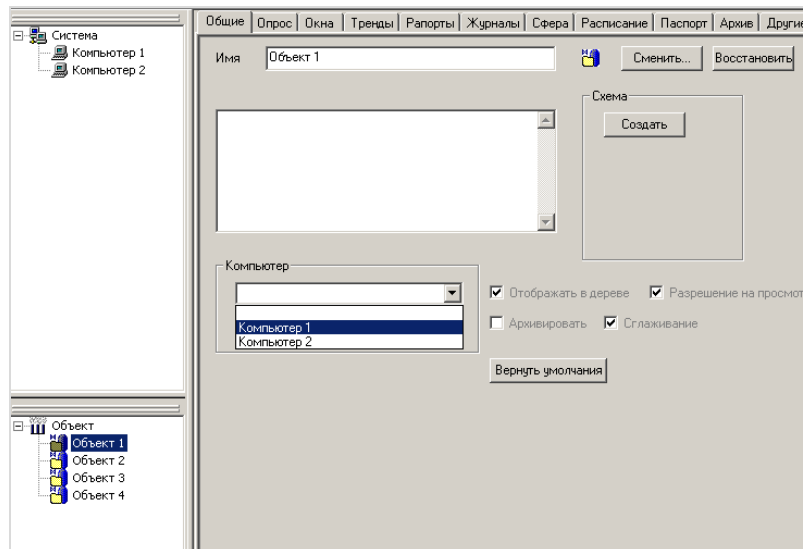


Для остальных компьютеров, созданных в дереве системы, сетевые настройки проводятся аналогично примеру выше.

- ❖ **Важно!** Для правильной работы MasterSCADA в сети, необходимо отключить брандмауэр Windows, или другой firewall, используемый системой. Необходимо также сделать общий доступ по сети к папкам проекта на обоих компьютерах.

Распределение объектов по узлам сети:

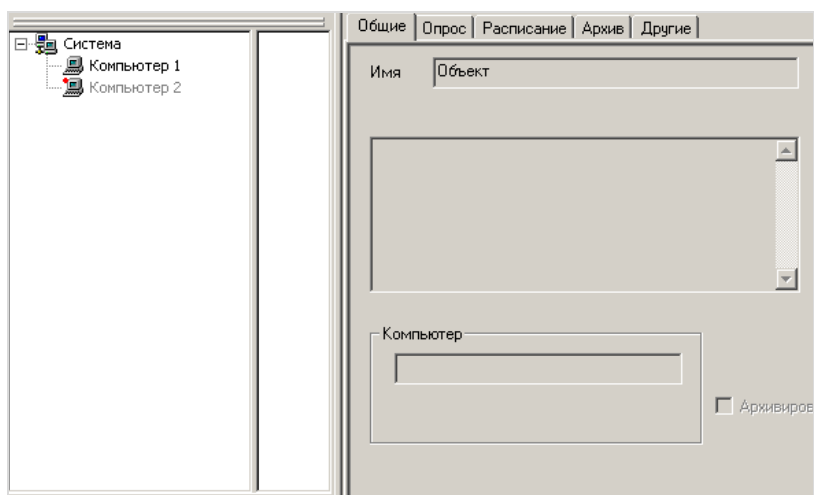
- Каждому компьютеру или контроллеру (если есть поддержка программирования из-под MasterSCADA) указывается, какой объект (объекты) на нем будет использоваться;



- Следующий шаг - это проведение операции экспортирования проекта и копирования его на все компьютеры сети.

Запуск системы:

После старта проекта появление красного кружка около символа компьютера или другого узла сети указывает на то, что сетевые настройки компьютера или контроллера были введены неверно, или компьютер был отключен. Необходимо еще раз проверить правильность настроек.



- ❖ **Важно!** При работе по сети MasterSCADA использует протокол TCP/IP порт 3675. Технология .Net Remoting.

Генератор отчетов

Редактор *MasterReport* предназначен для создания отчетов на основе данных из архива *MasterSCADA* или других источников. Отчет создается из отдельных частей – контейнеров-секций, которые называются *бэндами*. Бэнд не отображается в готовом отчете – задача бэндов выводить расположенные на нем компоненты в определенном месте, тем самым определяя порядок вывода содержащихся на нем элементов. Отчеты в генераторе отчетов строятся из *компонентов*. Компонент - это объект, который размещается на странице. Компоненты обладают самой разной функциональностью. Все компоненты можно разделить на пять основных групп:

- Страницы;
- Секции;
- Контейнеры;
- Компоненты для отображения информации;
- Элементы управления.

Комбинация компонентов из этих групп и образует отчет. *Компонент* обладает набором свойств, событий и методов. Изменяя свойства и задавая события компонента, можно изменять поведение компонента. Все свойства и события сгруппированы в категории:

- **Категория Расположение.** Положение компонента на странице определяется координатами X, Y его левого верхнего угла. Координаты X, Y и размер компонента описываются свойствами: *Лево*, *Верх*, *Ширина* и *Высота*. Значения задаются в текщей единице измерения;
- **Категория Редактирование.** Название компонента задается свойством *Наименование* - это программное имя компонента. Если в свойстве *Псевдоним* указывается какой либо текст, то везде, вместо наименования компонента будет отображаться этот текст. Свойство *Глобализация* используется при построении локализованных отчетов. Значение *Да* свойства *Замок* фиксирует положение и размер компонента. Рекомендуется использовать кнопку на панели инструментов *Дизайн* для управления свойством *Замок* выделенных компонентов. Свойство *Связь* позволяет привязать компонент к контейнеру, в котором находится компонент. Если оно установлено в значение *Да*, то компонент не может переместиться в другой контейнер. Например, если привязанный компонент перетащить с одной секции на другую, привязанный компонент все равно будет печататься вместе с первой секцией. Рекомендуется использовать кнопку на панели инструментов *Дизайн* для управления свойством *Связь* выделенных компонентов;
- **Категория Поведение.** Свойство *Доступность* позволяет включить обработку компонента при построении отчета. Если свойство равно *Нет*, то компонент будет полностью проигнорирован. Свойство *Печатать*, принимающее значение *Да* или *Нет*, определяет, будет ли компонент выведен на печать. Значение *Нет* указывает, что компонент не будет выведен на печать, при этом в окне предварительного просмотра он будет отображен. Значение *Да* указывает, что компонент, отображаемый в окне предварительного просмотра отчета, будет выводиться на печать. Свойства *Может расти* и *Может сжиматься* позволяют автоматически управлять изменениями размеров компонента в зависимости от его содержимого. Свойство *Режим смещения* позволяет избежать наложения компонентов при использовании средств автоматического изменения размеров компонента.

Выражения

Выражение – это сочетание математических и логических операторов, констант, функций, имен полей, элементов управления и свойств, в результате обработки которого возвращается единственное строковое значение. Получившееся значение сохраняется и используется в дальнейшей работе. Выражения, наиболее часто встречающиеся в генераторе отчетов – это текстовые выражения. Этот тип выражений используется для получения текста, который затем будет выведен на печать. Текстовые выражения всегда преобразуются в строку.

Наименование свойства, в котором записывается выражение	Свойство, в которое записывается результат расчета выражения	Описание
Text	TextValue	Используется для получения текста для вывода на печать. Всегда преобразуется в строку.
Bookmark	BookmarkValue	Используется для вычисления значения закладки.
Hyperlink	HyperlinkValue	Используется для вычисления строки гиперссылки.
Tag	TagValue	Используется для вычисления значения поля <i>Тэг</i> .
Tooltip	TooltipValue	Используется для вычисления строки для вывода в окне предварительного просмотра.
Checked	CheckedValue	Используется для вычисления строки для вывода в окне предварительного просмотра.
Filter	-	Используется для вычисления фильтра для секции <i>Данные</i> .
Condition	-	Используется для вычисления условия группировки в секции <i>Заголовок группы</i> .
Code	CodeValue	Используется для вычисления кодовой строки для вывода штрих-кода.

Текст в выражениях

Текстовые объекты являются простейшими выражениями:

- МойТекст;
- 12345.

Каждое из двух вышеуказанных выражений состоит из произвольной последовательности символов. В выражениях такого типа никакие вычисления не производятся. Данные выражения используются для отображения простых строковых констант: названий отчетов, столбцов, ссылок и т.д.

Вычисление значений в выражении

Выражение, кроме простого текста может также содержать различные переменные, значения из полей баз данных, функции. Для этого в выражение можно включать участки программного кода. Код пишется на языке программирования отчета. Для выделения кода предназначено два символа – { и } (фигурные скобки). Символ { обозначает начало кода для вычисления значения. Соответственно, символ } обозначает конец кода для вычисления значения. Код между указанными символами вычисляется и выдается результат вычисления. В случае текстовых выражений результат вычисления автоматически преобразуется в строку. К примеру, следующее выражение: $\{1 + 2\}$, в результате вычисления возвращает: 3. В одном выражении допускается неограниченное количество вставок кода для вычисления значений. Например: *Стулья = {1 + 2}*, *Кресла = {2 + 3}* – после вычисления вернет следующий текст: *Стулья = 3, Кресла = 5*. В выражении не допускается вложенность кода для вычисления значений. К примеру, такое выражение неверно: К примеру: $\{1 + 2 + \{2 + 3\}\}$.

❖ **Важно!** В выражении не допускается вложенность кода для вычисления значений.

Многострочные выражения

Возможно ли в выражении ввести многострочный текст? Все что необходимо сделать, для создания многострочного выражения, это поставить перевод строки перед новой строкой (в редакторе кода перевод строки ставится нажатием клавиши *Enter*). К примеру:

Значение:
 $\{1+2\}$

После расчета это выражение выведет на печать следующие строки:

Значение:
3

Т.е. текст будет содержать две строки. Каких либо ограничений на количество строк генератор отчетов не накладывает. Многострочные выражения не имеют никаких ограничений на использование кода для расчета значений. Действуют те же правила, что и для однострочных выражений. Однако при создании многострочных выражений следует учитывать, что код в выражениях нельзя разбивать переводом строки. К примеру:

Значение: {1 + 2}

Данное выражение неверное. Правильной будет следующая запись: Значение: {1 + 2} или:

Значение:
{1 + 2}

Использование переменных из словаря

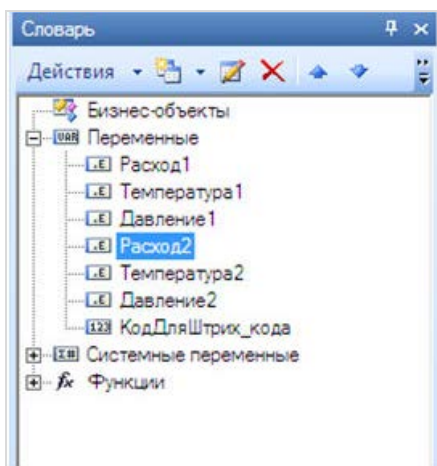
В выражениях можно использовать переменные созданные в словаре. Для этого в выражении просто указывается наименование переменной, к примеру: {МояПеременная}, вместо наименования переменной будет использовано ее значение.

- ❖ **Важно!** В выражении допускается использовать только имена переменных. Псевдонимы использовать не допускается.

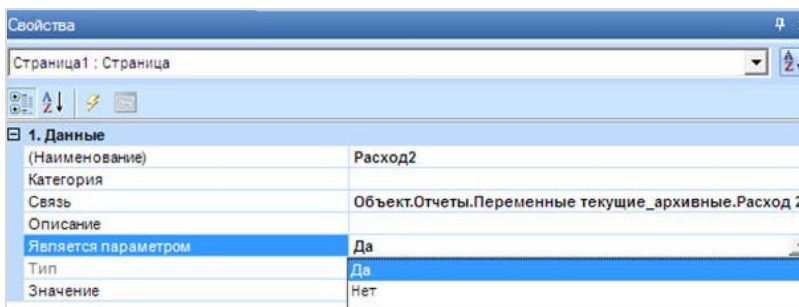
Переменные можно использовать в расчетах. К примеру: Значение = {МояПеременная + 10}, если переменная МояПеременная равна 15, это выражение вернет такую строку: Значение = 25.

- ❖ **Важно!** Если языком программирования отчета является С#, то имеет значения регистр написания наименований переменных. Если языком программирования является VB.Net, то регистр написания не имеет значения.

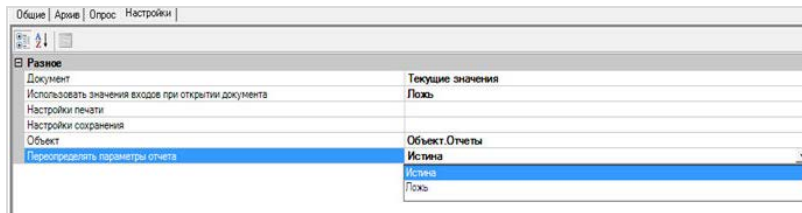
Использовать переменные из словаря можно ещё и по-другому. Переменные в словаре можно отметить как параметры.



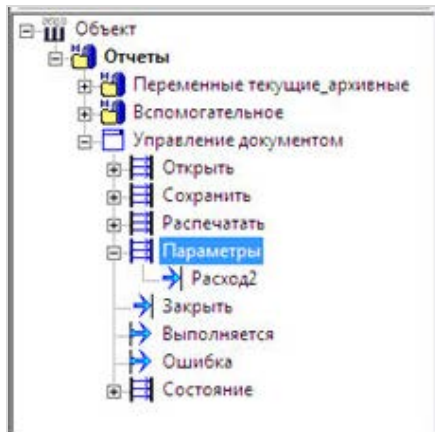
Переменные, находящиеся в словаре, имеют свойство *Является параметром*. Если данное свойство установить в значение *Да*, то переменная считается параметром:



Если в ФБ *Управление документом* в качестве документа выбран отчет, то после установки свойства *Является параметром* в значение *Истина* для переменной данного отчета в настройках ФБ *Управление документом* появляется свойство *Переопределять параметры отчета*:



Если данному свойству поставить значение *Истина*, то у ФБ появится группа входов *Параметры*, содержащие все параметры выбранного отчета.



Возможно использовать несколько ФБ *Управление документом*, чтобы в режиме исполнения вызывать один отчет с разными значениями переменных. В основном, это нужно для переменных начала и конца периода, за который создан отчет.

Использование полей из источников данных

В выражениях можно использовать значения из источников данных. Для этого используется ссылка в выражении на поле из источника данных. Ссылка представляет собой строковое представление поля. Сначала указывается наименование источника данных, затем через точку наименование поля: *{ИсточникДанных.Поле}*.

К примеру, выражение: *{Заказчики.НазваниеКомпании}* вернет наименование компании. Если источник данных имеет родительские связи с другими источниками данных, возможно вставить поля из родительского источника данных. В этом случае после названия источника данных через точку идет наименование связи. Затем через точку наименование поля: *{ИсточникДанных.КатегорияСвязи.Поле}*. К примеру: *{Products.ParentCategories.CategoryName}*, где:

- *Products* – это наименование источника данных;
- *ParentCategories* – это наименование связи, которой в данном случае связаны два источника данных;
- *Products* – список продуктов, и *Categories* – список категорий этих продуктов;
- *CategoryName* – наименование колонки в источнике данных *Categories*.

В результате вычисления выражение вернет наименование категории для продукта.

- ❖ **Важно!** В выражении допускается использовать только имена источников данных, связей, полей. Псевдонимы использовать не допускается.

Генератор отчетов не накладывает каких либо ограничений на количество переходов по связям. Т.е. возможно обратиться к колонке через две связи, через три. К примеру: *{OrderDetails.ParentProducts.ParentCategories.CategoryName}*.

в этом выражении:

- *OrderDetails* – это наименование источника данных;
- *ParentProducts* – наименование связи между источником данных *OrdersDetails* и *Products*;
- *ParentCategories*. – наименование связи между источником данных *Products* и источником данных *Categories*;
- *CategoryName* – поле в источнике данных *Categories*.

Как видно из примера значение поля *CategoryName* было получено при помощи обхода связей из источника данных *OrderDetails* к источнику данных *Categories*. При этом не было прямых обращений к источнику данных *Categories*.

- ❖ **Важно!** Если языком программирования отчета является C#, то имеет значения регистр написания наименований источников данных, связей, полей. Если языком программирования является VB.Net, то регистр написания не имеет значения.

Системные переменные

Генератор отчетов предлагает к использованию в выражениях **Системные переменные**. Системные переменные - это переменные, которые сообщают, какую либо информацию о текущем состоянии отчета. Доступные системные переменные:

- *Line* – возвращает номер текущей строки. Используется для нумерации строк в отчетах. Нумерация начинается с 1. Нумерация для каждой группы происходит отдельно;
- *LineThrough* – возвращает сквозной номер линии. В отличие от *Line*, возвращает номер линии с самого начала отчета, без учета группировок отчета. Нумерация начинается с 1;
- *PageNumber* – возвращает текущий номер страницы. Нумерация страниц начинается с 1. Используется для нумерации страниц;
- *TotalPageCount* – возвращает общее количество страниц в отчете;
- *PageNofM* – возвращает строку по следующему шаблону: *Страница {НомерТекущейСтраницы} из {ОбщееКоличествоСтраниц}*. Переменная используется для нумерации страниц с одновременным отображением общего количества страниц.
- *Column* – возвращает текущий номер колонки. Используется для нумерации колонок в отчете;
- *Today* – возвращает текущую дату;
- *Time* – возвращает текущее время;
- *ReportName* - возвращает наименование отчета;
- *ReportAlias* - возвращает псевдоним отчета;
- *ReportAuthor* - возвращает имя автора отчета;
- *ReportChanged* - возвращает дату когда был последний раз изменен отчет;
- *ReportCreated* - возвращает дату когда был создан отчет;
- *ReportDescription* - возвращает описание отчета.

Использование свойств компонентов

В выражениях допускается использовать свойства компонентов. При этом синтаксис полностью соответствует синтаксису C# или VB.Net. *{Компонент.Свойство}*. К примеру, для того чтобы в выражении обратиться к наименованию компонента: *{МойКомпонент.Имя}*. Если необходимо указать в выражении результат расчета другого выражения, то следует использовать не само свойство выражения, а свойство, в которое записывается результат. К примеру, значения гиперссылки: *{МойКомпонент.Гиперссылка}*

Функции в выражении

Генератор отчетов не имеет встроенных функций, но благодаря тесной интеграции с *Net Framework*, это не является проблемой. Возможно использовать любые функции доступные в *Net Framework*. К примеру, для строковых значений:

```
{Trim(" МояСтрока ")}  
{Trim(МояПеременная)}  
{ToUpperCase("Тест")}  
{Length(МояПеременная)}
```

Для числовых значений:

```
{Round(МояПеременная, 2)}  
{Sqrt(МояПеременная)}  
{ToString(МояПеременная) + " "}
```

Выражения с условием

По умолчанию генератор отчетов не допускает условий в выражении. Однако при использовании языка программирования C# можно использовать тернарную операцию. Вид тернарной (*ternary*) операции: {Условие ? Значение1 : Значение2}.

Если условие возвращает значение *Истина*, то выражение вернет *Значение 1*. Если нет, то *Значение 2*. К примеру: {Запас.Count > 0 ? Запас.Count : "Пусто"}.

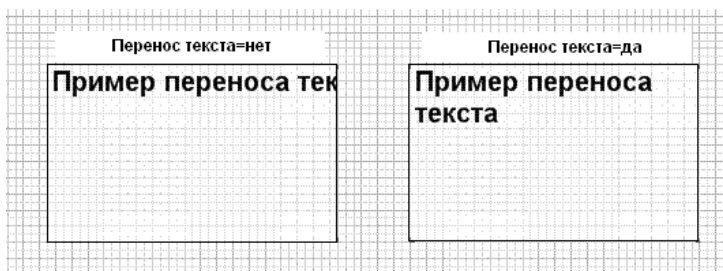
Отображение текстовой информации

Текст – это одна из основных частей любого отчета. Для вывода текста в генераторе отчетов используется несколько компонентов: Текст, Rtf - текст. Текстовые компоненты обладают богатыми возможностями оформления. Для настройки доступны множество параметров - позиция и размеры, рамка, заливка, выравнивание, шрифт и. д. Кроме оформления текстовые компоненты предоставляют множество параметров вывода текста.

Параметры вывода текста

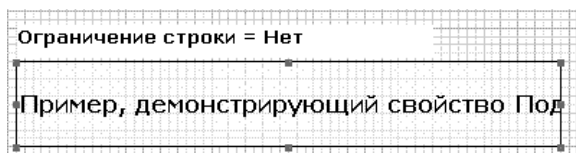
Для текстовых компонентов существует достаточно большое количество свойств. Изменяя эти свойства можно влиять на внешний вид выводимого текста. В этом разделе описывается множество самых разных по своему назначению свойств. Большая часть параметров специфична только для текстовых компонентов.

Многострочный текст. По умолчанию, если текст не уместится в одну строку, то он будет обрезан. Если требуется, чтобы текст располагался в несколько строк, то необходимо включить автоматический перенос текста. Для этого необходимо установить свойство компонента Текст *Перенос текста* в *Да*. При переносе текста на новую строку учитываются вертикальное и горизонтальное выравнивание.



Добавление в конец обрезанной строки многоточия. Если строка не помещается полностью в ограничивающий прямоугольник, то с помощью свойства *Подгонка* можно автоматически добавить многоточие. Свойство *Подгонка* может принимать следующие значения:

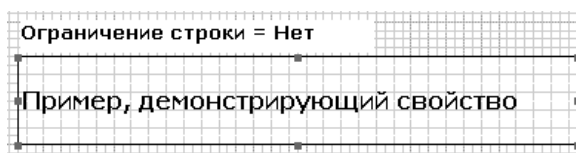
- *Не подгонять* - строка обрезается строго по границе ограничивающего прямоугольника (или по последнему видимому слову, если это многострочная строка);



- *Символ* - строка обрезается за последним видимым символом;



- *Слово* - строка обрезается за последним видимым словом;



- **Символ многоточие** – последние нескольких символов заменяются на многоточие;



- **Слово многоточие** - за последним, полностью видимым словом, добавляется многоточие;



- **Путь многоточие** - средняя часть строки заменяется на многоточие так, чтобы начало и конец строки были видимы.



Предотвращение появления неполных строк. Часто возникает необходимость выводить текст так, чтобы не было обрезанных по вертикали строк снизу. Если установить свойство *Ограничение строки*, то будут выводиться только полные строки. Отсутствие дополнительной строки может поменять перенос слов.

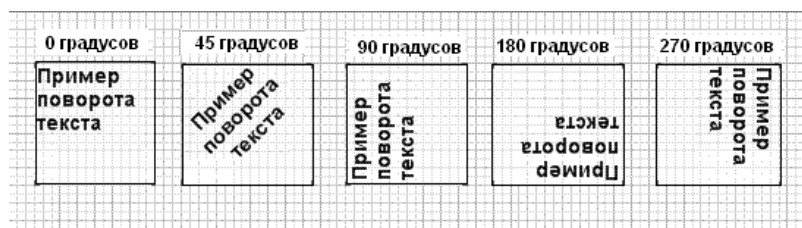


Если установить значение свойства **Линии подчеркивания** в *Да*, то каждая строка компонента *Текст* будет подчеркнута.

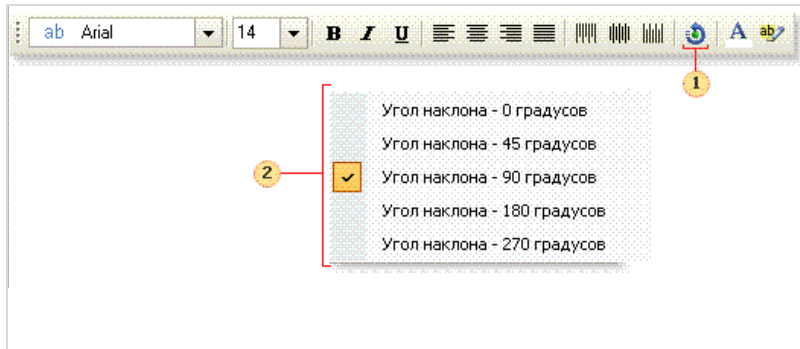


Как сделать, чтобы компонент *Текст* при увеличении размера по вертикали увеличивал его только до какого-то максимального количества строк? Используйте свойство **Максимальное число строк**. По умолчанию это свойство равно 0 и компонент может увеличиваться по вертикали неограниченно (в пределах страницы). Если установить значение этого свойства равным, например 5, то при увеличении размера по вертикали компонента он увеличится максимум до 5 строк в высоту.

Установить **Угол наклона** текста можно при помощи свойства *Угол*. Угол наклона текста задается в градусах против часовой стрелки.




Для установки стандартных значений угла наклона текста (0, 45, 90, 180 и 270 градусов) рекомендуется использовать кнопку *Угол*, расположенную на панели инструментов *Форматирование*.



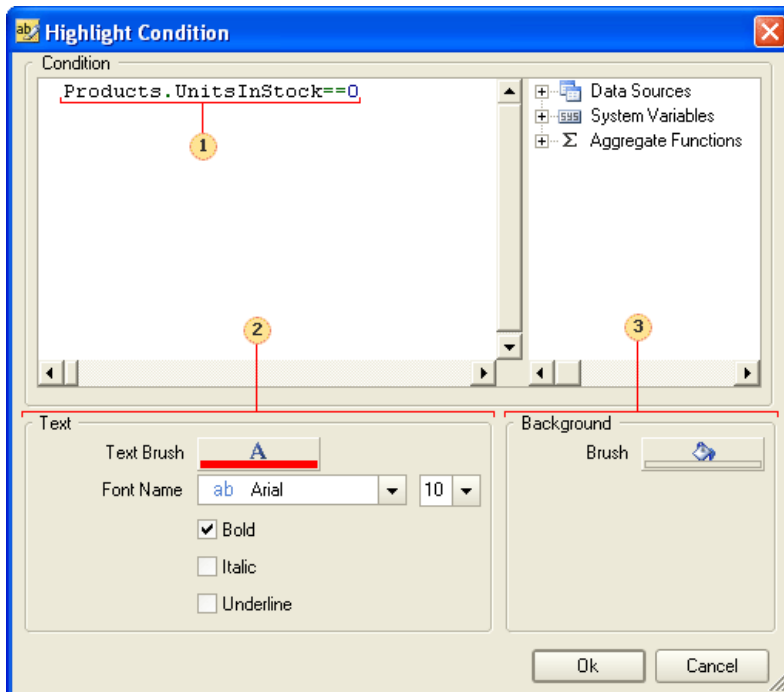
Цифрой 1 обозначена кнопка вызова панели стандартных значений для угла поворота текста. Цифрой два панель стандартных значений угла поворота текста.

Объединение повторяющихся значений. Во многих отчетах возникает необходимость объединить несколько компонентов *Текст* в один, если эти компоненты содержат повторяющиеся значения. Для того чтобы компонент *Текст* объединял повторяющиеся значения, необходимо установить значение свойства *Обработка дубликатов* в *Да*.

Условное выделение позволяет изменить оформление компонента *Текст* в зависимости от определенного условия. Настроить условное выделение можно воспользовавшись свойством *Условия* или кнопкой  на панели инструментов *Форматирование*.



Кнопка один вызывает окно условного выделения. Параметры условного выделения задаются в специальном окне. В окне условного выделения указываются условие и параметры оформления компонента.



В левом столбце - если условие верно, то оформление компонента меняется на указанное в параметрах. Группа *Текст* содержит параметры оформления выделенного компонента. Группа *Фон* содержит параметры заполнения фона выделенного компонента. Для того чтобы удалить условное форматирование просто удалите в редакторе весь текст условия. *Условием* может быть любое выражение. В случае если условие, верно, то оформление компонента *Текст* будет изменено. Для настройки доступны свойства *Цвет текста*, *Изменить шрифт* и *Цвет фона*. Например, если запасы на складе продукта равны нулю, то строку можно выделить красным цветом. Для этого в условии необходимо написать: *Products.UnitsInStock = 0*. Для *Цвет фона* указываем красный цвет, для *Цвет текста* – белый цвет.

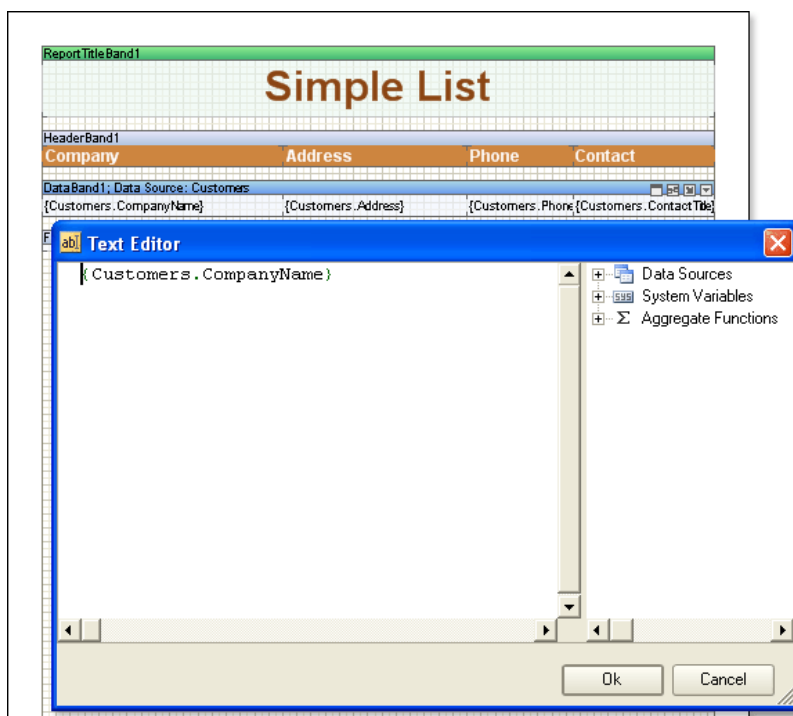
Meat/Poultry	Wimmers gute Semmelknödel	20 bags x 4 piece	33,25p.	22,00
	Alice Mutton	20 - 1 kg tins	39,00p.	0,00
	Mshi Kobe Niku	18 - 500 g pkgs.	97,00p.	29,00
	Pâté chinois	24 boxes x 2 pies	24,00p.	115,00
	Perth Pasties	48 pieces	32,80p.	0,00
	Thüringer Rostbratwurst	50 bags x 30 sau:	123,79p.	0,00
	Tourtière	16 pies	7,45p.	21,00

Игнорирование нулевых значений. Часто, при выводе числовой информации на печать, требуется игнорировать нулевые значения. Т.е. не выводить их на печать совсем. Установите значение свойства *Скрывать нулевые значения* в *Да*, и компонент *Текст* не будет печатать нулевые значения.

Свойство **Выводить в** компонента *Текст* предназначено для вывода сообщения одновременно сразу в двух текстовых компонентах. Выводимое сообщение указывается в первом текстовом компоненте. Далее, в этом текстовом компоненте, в свойстве *Выводить в* указывается второй текстовый компонент, в котором будет продолжен вывод сообщения. Сообщение будет продолжаться, выводится во втором компоненте, только если в первом компоненте оно не поместилось целиком. При этом стоит учитывать, что в первом компоненте будет выведено только целое количество, полностью видимых по вертикали строк. Во втором компоненте сообщение продолжает выводиться ровно с той позиции в сообщении, с которой был прекращен вывод в первом компоненте. Обязательным условием правильной работы функции является то, что сначала должен быть создан первый компонент, а затем второй. Если же порядок создания компонентов был другой, то можно воспользоваться командами перемещения порядка компонентов. Свойство *Выводить в* позволяет работать только с компонентами, которые размещены на одном уровне – например на секции.

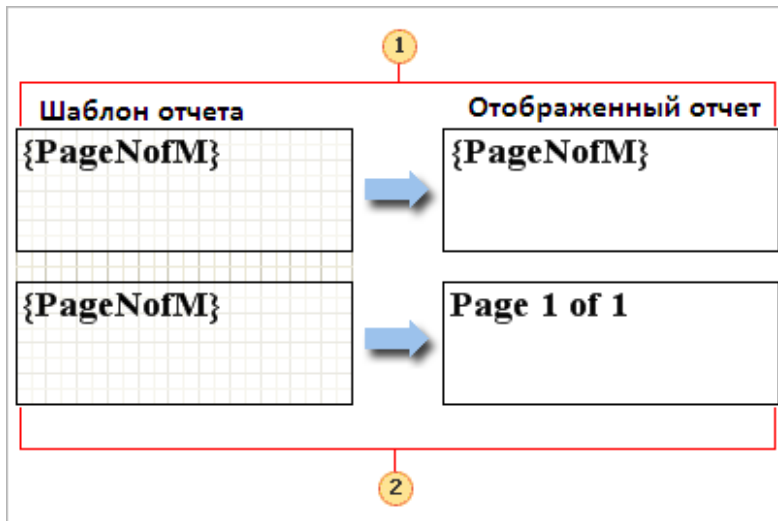
Выражения в тексте

Весь текст, который отображается в отчете, формируется при помощи текстовых выражений. *Текстовые выражения* ни чем не отличаются от других выражений. Выражение задается в свойстве *Текст*.



Текстовое выражение имеет одно отличие от других выражений – на окончательный результат оказывает влияние *форматирование текста*. Кроме того, для текстовых выражений, возможно, задавать дополнительные параметры которые влияют на окончательный результат.

Для **Вывода только текста без учета выражений** установите значение свойства *Только текст* в *Да*, и все выражения будут выведены как текст. Никаких расчетов производится не будет.



В верхней части рисунка выше свойство *Только текст* установлено в *Да*. Текст выводится как есть, без обработки выражений. В нижнем - свойство *Только текст* установлено в *Нет*. Текст выводится с обработкой выражений.

По умолчанию выражения, которые встречаются генератору отчетов в тексте, обрабатываются сразу. Но иногда возникает необходимость **Обработать выражения после построения отчета**. Например, во время построения отчета производится расчет, какой-то переменной. Результат этого расчета будет известен только в конце построения отчета, а вывести его нужно, к примеру, на каждой странице.

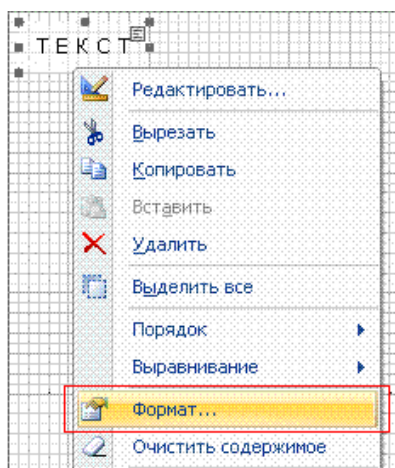
- ❖ **Важно!** *Выражения компонентов Текст, у которых установлено это свойство в Да обрабатываются в конце построения отчета.*

Чтобы генератор отчетов рассчитал и вывел выражение после построения отчета этого установите значение свойства *Обработать* в конце компонента *Текст* в *Да*.

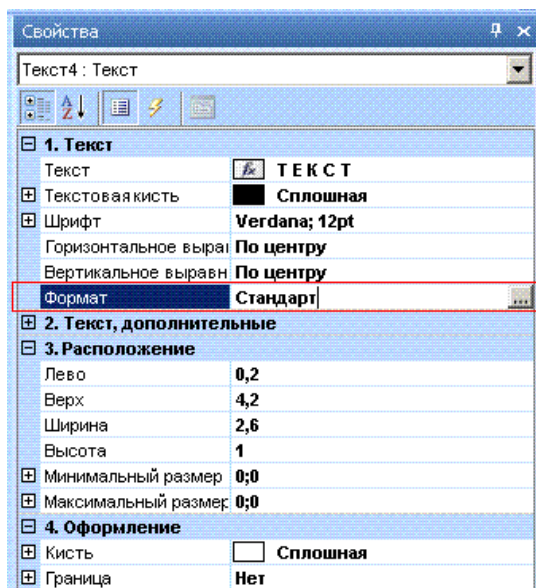
- ❖ **Важно!** *В случае обработки содержимого текстового компонента в конце отчета, генератор отчетов не может корректно выполнить определение размера компонента в момент его вывода. Поэтому автоматическое изменение размеров компонента будет работать неверно.*

Форматирование текста

Форматирование текста - это отображение информации в специальном виде, группирование и отображение представления данных согласно указанному шаблону. Генератор отчетов содержит все необходимые инструменты для форматирования выводимой информации. Основным инструментом форматирования значений перед выводом является *Формат текста*. Этот инструмент представляет собой диалоговое окно, которое позволяет произвести настройку ряда параметров форматирования. *Диалог форматирования* текста вызывается из контекстного меню, которое появляется при нажатии правой кнопкой мыши на текстовых компонентах, которые поддерживают форматирование.



Также диалог форматирования можно вызвать при помощи свойства *Формат* панели СВОЙСТВ.

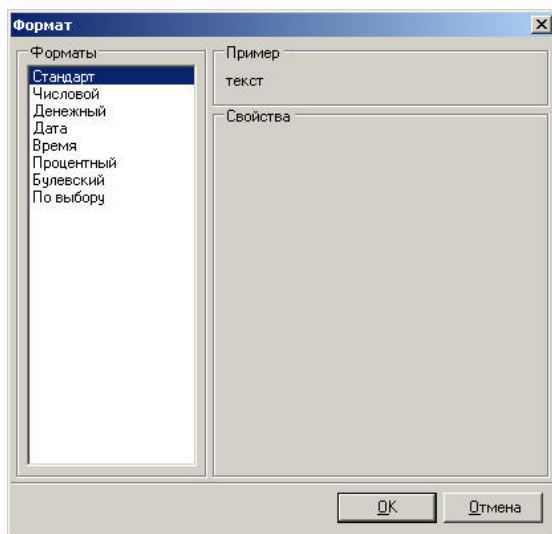


Окно форматирования текста разделено на три части. В левой части можно выбрать тип форматирования. Всего предусмотрено несколько типов отображения текста:

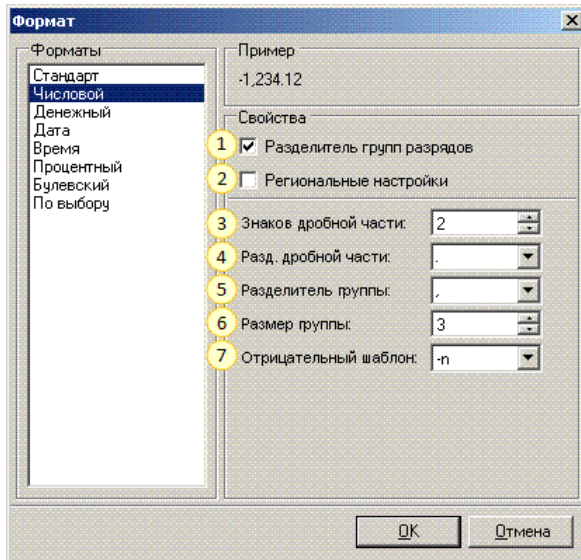
- *Стандарт* - отображение данных без форматирования;
- *Числовой* — форматирование чисел;
- *Денежный* — форматирование валюты;
- *Дата* — форматирование данных связанных с датой;
- *Время* — форматирование данных, которые представляют время;
- *Процентный* — форматирование данных, которые классифицируются как проценты;
- *Булевский* — форматирование булевых типов данных;
- *По выбору* — пользовательский тип отображения данных.

В правой части выводится пример формируемого текста по тем правилам, которые выбраны. В правую нижнюю часть можно выставить настройки форматирования.

Тип форматирования **Стандарт** используется для отображения как текстовых, так и числовых значений произвольного типа. При отображении никакого форматирования не производится.



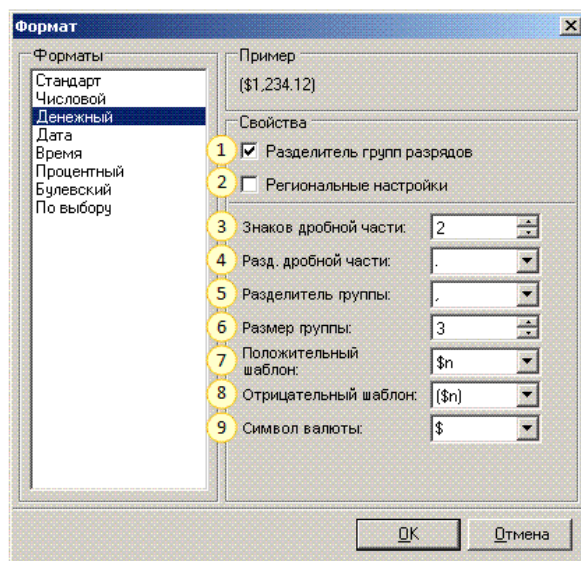
Для **Отображения числовых значений**, рекомендуется использовать числовой формат. Для вывода денежных значений используется также денежный формат.



Функции элементов управления:

- При использовании *Разделителя групп*, числовые величины разделяются на группы цифр;
- При использовании *Локальных установок*, числовые величины форматируются согласно текущим установкам операционной системы;
- *Знаков в дробной части* - количество десятичных цифр, которые используются при форматировании числовых величин;
- *Разделитель дробной части* - строка, которая используется как десятичный разделитель при форматировании числовых величин;
- *Разделитель группы* - строка, которая используется как разделитель групп при форматировании числовых величин;
- *Размер группы* - количество цифр в каждой группе при форматировании числовых величин;
- *Отрицательный шаблон* - шаблон форматирования отрицательных значений.

Для Форматирования денежных значений используйте формат **Денежный** для вывода денежных значений. Денежный формат может быть использован также и для вывода обычных чисел.

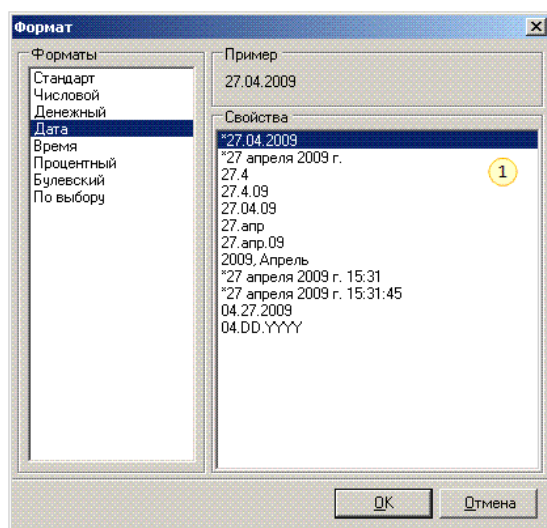


Функции элементов управления:

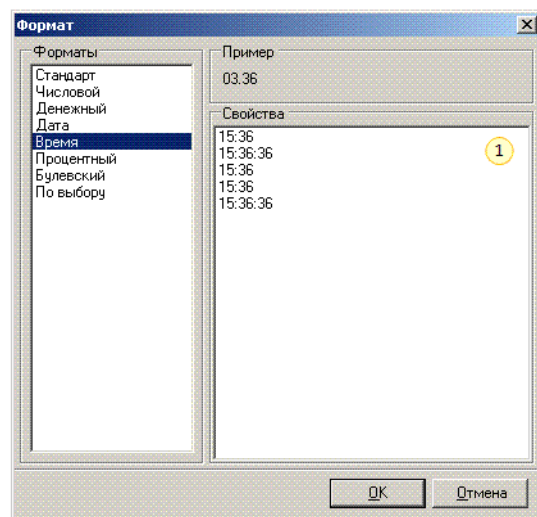
- *Разделитель групп разрядов* - при использовании разделителя групп, денежные значения разделяются на группы цифр;
- *Региональные настройки* - при использовании локальных установок, денежные значения форматируются согласно текущим установкам операционной системы;

- *Знаков в дробной части* - количество десятичных цифр, которые используются при форматировании денежных значений;
- *Разделитель дробной части* - строка, которая используется как десятичный разделитель при форматировании денежных значений;
- *Разделитель группы* - строка, которая используется как разделитель групп при форматировании денежных значений;
- *Размер группы* - количество цифр в каждой группе при форматировании денежных значений;
- *Положительный шаблон* - шаблон форматирования положительных значений;
- *Отрицательный шаблон* - шаблон форматирования отрицательных значений;
- *Символ валюты* - символ, который будет использоваться в качестве указания валюты.

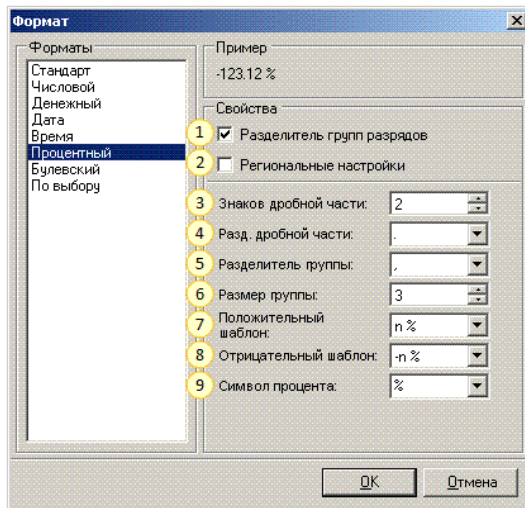
Тип форматирования **Дата** предназначен для отображения даты. Формат вывода даты выбирается из набора предусмотренных форматов - короткий формат вывода дат, длинный формат и т.д. В применяемых форматах, за исключением помеченных звездочкой(*), порядок элементов не меняется.



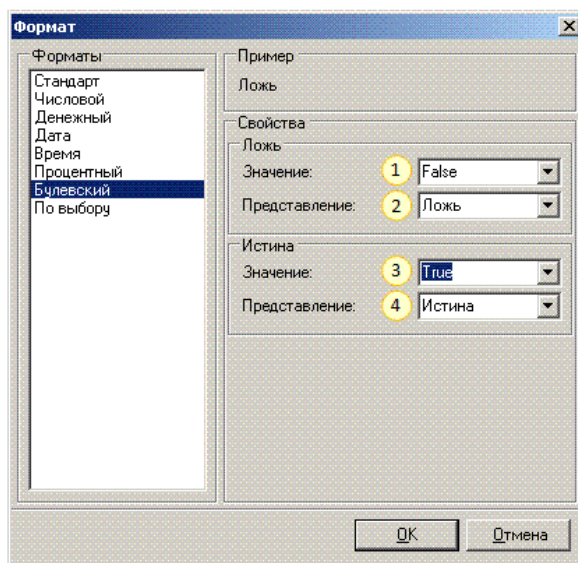
Тип форматирования **Время** предназначен для отображения времени. Для данных типа **Время** можно выбрать варианты отображения времени: короткий тип отображения, и длинный тип отображения (с секундами).



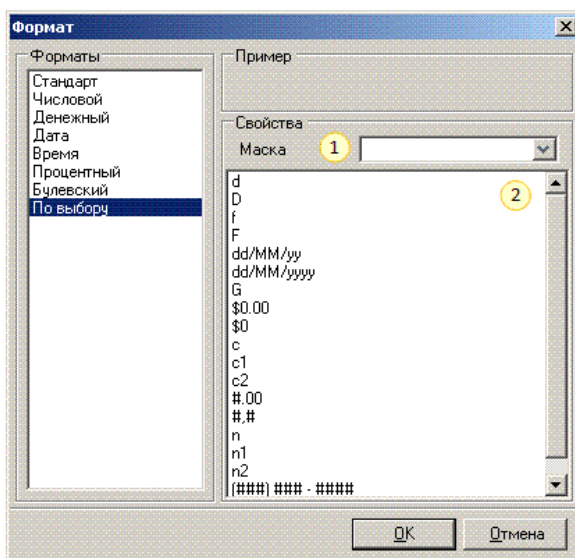
Формат **Процентный** используется для отображения значений с процентом. При форматировании значение умножается на 100 и выводится со знаком процента. Настройки свойств схожи с **Денежным** форматом.



Формат **Булевский** используется для форматирования значений булевого типа.



Пользовательский тип отображения **По выбору**. Этот тип форматирования позволяет настроить формат вывода данных (в поле *Маска*).



Маска - строка форматирования значения задаваемая пользователем. *Предопределенные значения* - список predefined значений для строки форматирования

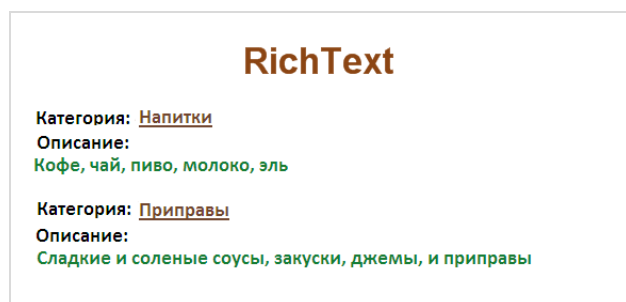
Инструмент **Формат текста** позволяет произвести форматирование выводимых значений с учетом множества параметров и разными вариантами. Но этот инструмент имеет один не-

достаток - форматирование действует целиком на весь текстовый объект. К примеру, если текстовый компонент используется для отображения даты, то произвести форматирование очень легко. Если требуется произвести форматирование только определенного значения из выражения или выполнить форматирование нескольких значений в одном выражении рекомендуется воспользоваться методом *string.Format*. Используя этот метод можно выполнить практически все те же виды форматирования, что и при помощи инструмента *Формат текста*. Однако метод *string.Format* обладает гораздо большей гибкостью. К примеру, для того чтобы отформатировать значение как денежное, используется спецификатор *C*: *Валюта: {string.Format("{0:C}", Value) }*, при значении равном *123.12* после вычисления выдаст следующую строку: *Валюта: \$123.12*.

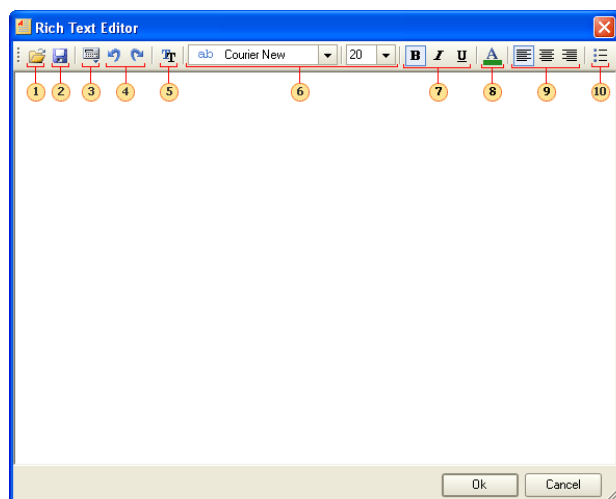
Метод *string.Format* также принимает более одного параметра для форматирования, к примеру: *Значение формат Денежный: {string.Format("value1 - {0:C}, value2 - {0: 1}", Value1, Value2) }*.

Отображение Rtf текста

Генератор отчетов позволяет включать в отчеты текст в формате *Rtf*. При этом не накладывается каких либо существенных ограничений. Компонент для работы с *Rtf* текстом называется *Rtf* текст. Этот компонент умеет автоматически изменять свои размеры в зависимости от размеров *Rtf* текста в нем, умеет обрабатывать выражения, поддерживает различные варианты оформления, обработку в конце построения отчета и т.д.



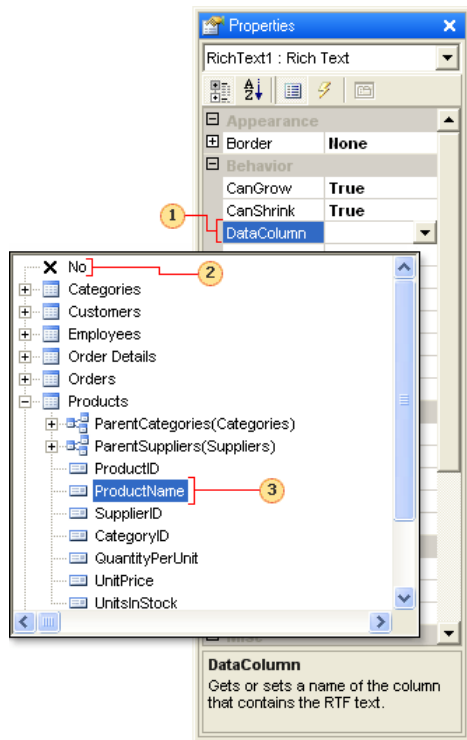
Для редактирования *Rtf* текста, компонент *Rtf* текст имеет специальный редактор. Этот редактор умеет загружать и сохранять *Rtf* текст, менять шрифт, размер, цвет, вставлять выражения и т.д. Используя этот редактор можно полностью отредактировать *Rtf* текст, не прибегая к сторонним редакторам. Редактор вызывается двойным щелчком мыши по компоненту *Rtf текст*. Кнопки редактора аналогичны кнопкам MS Word.



Rtf текст в компоненте *Rtf текст* рассматривается как выражение. Существенных отличий при работе с выражениями в *Rtf текст* компоненте от других текстовых компонентов нет. Правила написания и использования выражений полностью совпадает с правилами для выражений в текстовых компонентах. Но существует одна особенность. Форматирование не должно разрывать вставки кода для вычисления значений на несколько частей. Т.е. если, к примеру, в *Rtf тексте* необходимо вывести значение, которое рассчитывается при помощи вставки кода определенным цветом то необходимо чтобы этот цвет был установлен для всей вставки кода. Код в выражении начинается с символа *{* и заканчивается символом *}*.

Соответственно любое форматирование должно относиться целиком к коду начиная с символа { (включая этот символ) и заканчивая символом } (включая этот символ). К примеру: {Categories.CategoryName} – выражение является вставкой кода, и форматирование применено целиком к этой вставке кода. Это выражение будет работать правильно.

Кроме прямой работы с Rtf текстом Rtf текст умеет загружать Rtf текст из поля данных. Для этого предназначено специальное свойство *Колонка данных*. Все что необходимо, для того чтобы загрузить *Rtf текст* – это просто выбрать поле из дерева словаря данных. При построении генератор отчетов будет автоматически загружать Rtf текст, при каждом построении Rtf текст компонента.



Свойство *Колонка данных* служит для указания, из какого поля данных будет загружаться *Rtf текст*. Выбор нулевого узла означает, что загрузка *Rtf текста* из поля данных не используется. Выбранное поле - поле данных, из которого будет происходить загрузка *Rtf текста*.

Отображение графической информации

Кроме текстовых компонентов в отчетах также можно использовать графические компоненты. Генератор отчетов предоставляет следующие графические компоненты: *компонент Картинка*, *компонент Геометрия*, *компонент Штрих-код*. *Компонент Картинка* служит для отображения изображений. *Компонент Геометрия* служит отображения геометрических фигур. *Компонент Штрих-код* отображает штрих коды.

Отображение изображений

Для большей наглядности в отчеты часто добавляют изображения. Это могут быть изображения товаров, фотографии сотрудников и т.д. Также часто в отчетах выводятся логотипы фирм. Для вывода изображений в генератор отчетов используется *компонент Картинка*. Этот компонент поддерживает следующие типы изображений: *Bmp, Jpeg, Tiff, Gif, Png, Ico, Emf, Wmf*.

Растягивание изображения

При выводе изображений очень часто размеры изображения не совпадают с размерами компонента. В этом случае остается пустое, незаполненное изображением пространство. Также возникают ситуации, когда размеры изображение больше чем размеры компонента. В данных ситуациях необходимо чтобы *компонент Картинка* всегда растягивал размеры

изображения в размеры компонента. Для этого необходимо свойство *Растягивать установить* в Да.

После установки свойства *Растягивать* размеры изображения будут всегда соответствовать размерам компонента. При растягивании изображения могут нарушиться его пропорции. Для того чтобы заставить компонент картинка растягивать изображения, но при этом сохранять пропорции изображения, необходимо установить свойство *Отношение координат* в Да. После этого компонент картинка всегда будет сохранять пропорции изображения.

- ❖ **Важно!** *Свойство Отношение координат действует только когда включено растягивание изображения.*

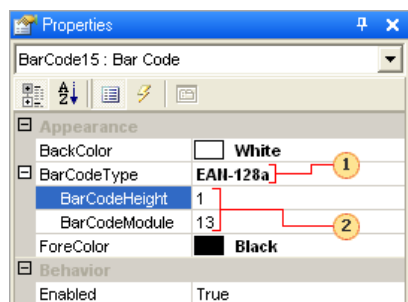
Отображение штрих-кодов

Генератор отчетов поддерживает различные типы штрих-кодов: *EAN-8, EAN-13, UPC-A, UPC-E, UPC-Sup2, UPC-Sup5, 2 of 5 Standard, 2 of 5 Interleaved, Code 39, Code 39 Extended, Codabar, Code 128a, Code 128b, Code 128c, EAN 128a, EAN 128b, EAN 128c* и пр. Для указания кода для отображения в штрих-коде используется специальное свойство Код.

1. Штрих-код	
Код	<i>Fr</i> 1234567890123
Тип штрих-кода	EAN-13
Код дополнения	
Высота	1
Модуль	13

Это свойство является выражением, поэтому в нем можно указывать не только строку с кодом но и выражения для расчета это кода. К примеру, код задан в виде строки: 1234567890123. Код из поля данных: *{Items.Code}*

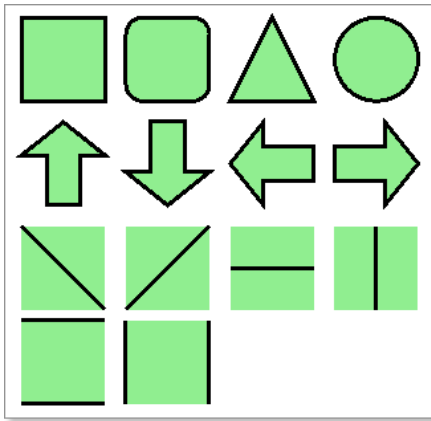
При использовании компонентов штрих-код следует учитывать, что изменение размеров компонента не влечет изменения размеров самого штрих-кода. Все штрих-коды отображаются строго по стандартам. Многие виды штрих-кодов или вообще не допускают изменение размеров или допускают в определенных рамках. Поэтому размеры указываются через дополнительные свойства. Вы можете увидеть эти свойства, если в панели *Свойства* развернете *Тип штрих-кода*. К примеру, штрих-код *EAN-128a* позволяет настраивать два параметра: масштаб отображения штрих-кода и количество модулей.



- ❖ **Важно!** *Изменение размеров компонента не влечет изменения размеров самого штрих-кода. Используйте дополнительные свойства.*

Отображение геометрических фигур

Во многих отчетах требуется выводить на печать геометрические фигуры. Для этого в генераторе отчетов рекомендуется использовать компонент *Геометрия*. Геометрическим фигурам можно указывать различное оформление – кисть заполнения, цвет границ и т.д. Этот компонент умеет вводить следующие типы геометрических фигур: *прямоугольник, прямоугольник со скругленными углами, треугольник, овал, стрела* и различные виды *линий*.

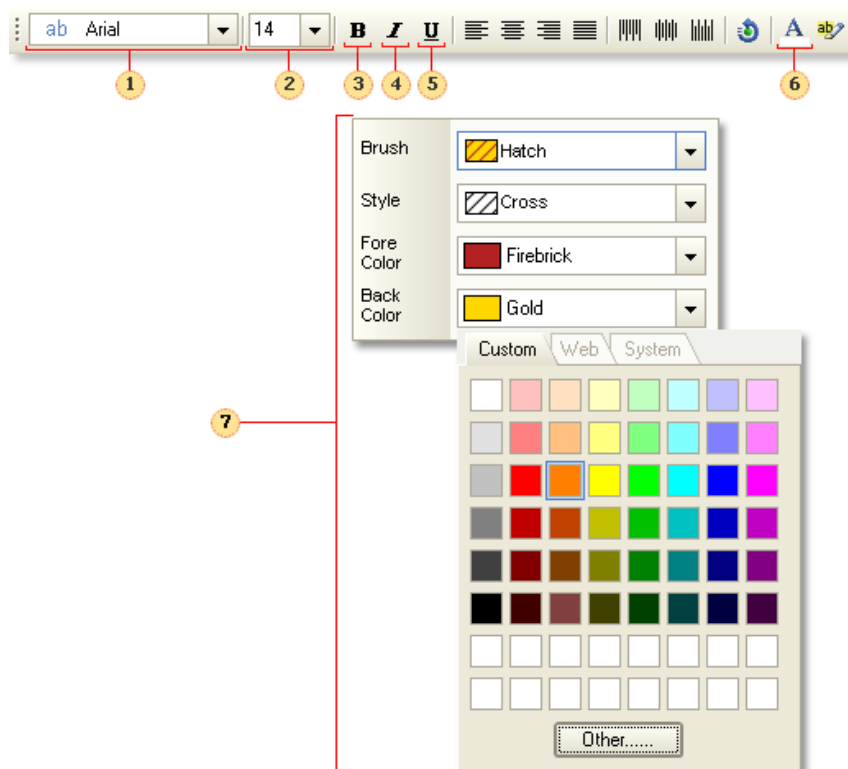


Оформление компонентов

Генератор отчетов предоставляет множество параметров для оформления внешнего вида компонентов. Доступны для изменения шрифт, размеры шрифта, цвет шрифта, фон компонента, рамки компонента, горизонтальное и вертикальное выравнивание.

Шрифт

Текст можно выводить, используя разные шрифты, разные размеры шрифтов, разные кисти для заполнения фона. Параметры шрифта можно изменить в свойстве *Шрифт*. Кисть выводимого текста настраивается в поле *Текстовая кисть*. Для настройки шрифта рекомендуется использовать кнопки панели инструментов *Форматирование*.



Шрифт - в выпадающем списке можно указать шрифт, который будет использоваться для вывода текста. Размер шрифта - поле позволяет установить размер шрифта выводимого текста. Также возможно выбрать значения из выпадающего списка. *Полужирное начертание* - кнопка управляет полужирным начертанием текста. *Наклонный шрифт* - кнопка управляет наклонным начертанием текста. *Подчеркнутый шрифт* - кнопка управляет подчеркнутым начертанием текста. *Кисть шрифта* - элемент управления позволяет задать кисть для закрашки выводимого текста. *Панель управления цветом шрифта* - панель управления кистью, предназначена для настройки кисти закрашки выводимого текста.

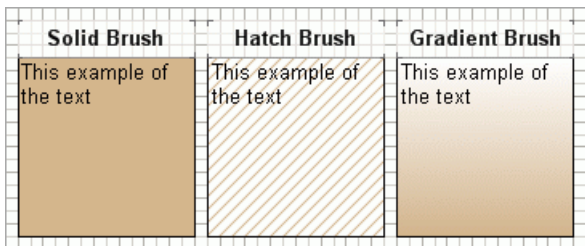
Фон

Для заполнения фона используются кисти. Для изменения заливки фона используется свойство *Кисть компонента* или кнопка на панели инструментов *Границы*.



Доступны три типа кистей:

- *Сплошная заливка* – фон компонента заполняется указанным цветом.
- *Заливка узором* – фон компонента заполняется узором. Дополнительно указывается цвет фона узора и цвет узора.
- *Градиентная заливка* – фон заполняется градиентным переходом цветов. Указывается цвет начала градиента, цвет конца градиента и угол градиентной заливки.



Рамка

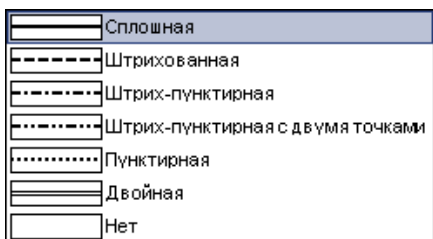
Многие компоненты имеют возможность отображать рамку. Рамка может иметь разную толщину, цвет и стиль. Также компонент может отображать тень. Для управления рамкой компонента используется свойство *Граница компонента* или панель инструментов *Границы*.



Рамки - кнопки управления рамкой. Используются для настройки и отображения границ рамки.

- - граница со всех сторон. После нажатия устанавливает всем выделенным компонентам границу со всех сторон;
- - нет границы. После нажатия убирает границу у всех выделенных компонентов;
- - граница с верхней стороны. После нажатия устанавливает или снимает всем выделенным компонентам границу с верхней стороны;
- - граница с левой стороны. После нажатия устанавливает или снимает всем выделенным компонентам границу с левой стороны;
- - граница с нижней стороны. После нажатия устанавливает или снимает всем выделенным компонентам границу с нижней стороны;
- - граница с правой стороны. После нажатия устанавливает или снимает всем выделенным компонентам границу с правой стороны.

Тень - Кнопка управления тенью. После нажатия включает или выключает (если включена) отображение тени компонента. Цвет рамки - элемент управления используется для выбора цвета рамки компонента. Стиль рамки - элемент управления используется для выбора стиля рамки



Горизонтальное выравнивание

Некоторые компоненты позволяют задать выравнивание своего содержимого относительно своих размеров по горизонтали. Например, компоненты *Текст*, *Картинка*. Выравнивание по горизонтали определяет внешний вид содержимого и может выполняться по левому краю, по правому краю, по центру или по ширине (только для текста). Для изменения выравнивания используется свойство компонента *Горизонтальное выравнивание*. Также выравнивание можно изменить при помощи кнопок на панели инструментов Форматирование.

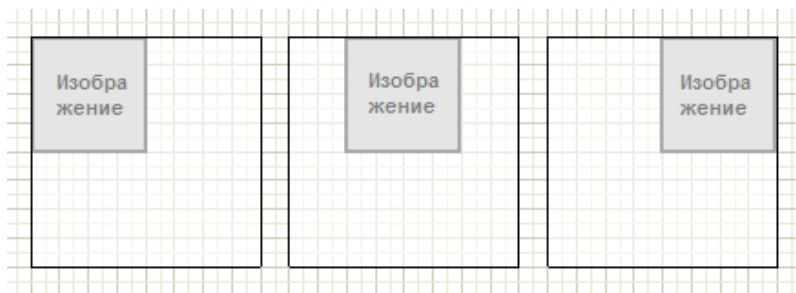


- *По левому краю* - содержимое компонента выравнивается относительно левой стороны. После нажатия устанавливает всем выделенным компонентам выравнивание по горизонтали к левой стороне;
 - *По центру* - содержимое компонента выравнивается по центру относительно левой и правой сторон. После нажатия устанавливает всем выделенным компонентам выравнивание по центру;
 - *По правому краю* - содержимое компонента выравнивается относительно правой стороны. После нажатия устанавливает всем выделенным компонентам выравнивание по горизонтали по правой стороне;
 - *По ширине* - содержимое компонента выравнивается одновременно относительно и левой и правой сторон. После нажатия устанавливает всем выделенным компонентам выравнивание по ширине.
- ❖ **Важно!** Выравнивание по ширине доступно только для компонентов, отображающих текст.

Горизонтальное выравнивание текста. В большинстве случаев текст выравнивается по левому краю. В случае выравнивания по ширине, текст выравнивается одновременно как по левому, так и по правому краю. Выравнивание текста по ширине позволяет получить ровные края текста по бокам.

Выравнивание по левому краю	Выравнивание по центру	Выравнивание по правому краю	Выравнивание по ширине
Пример выравнивания текста	Пример выравнивания текста	Пример выравнивания текста	Пример выравнивания текста

Горизонтальное выравнивание изображение. Для управления выравниванием по горизонтали для компонента *Картинка* используются те же самые элементы управления, что и для компонента *Текст*. *Изображения* выравнивается только в случае, если свойство *Растягивать* установлено в *Нет*. Иначе параметры выравнивания будут игнорироваться.



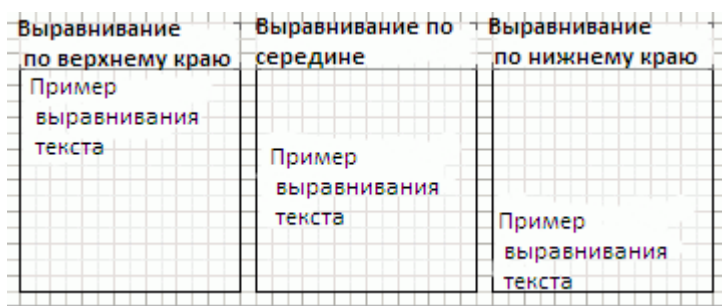
Вертикальное выравнивание

Выравнивание по вертикали определяет положение содержимого относительно верхней и нижней границы компонента. Вертикальное выравнивание может быть установлено по верхнему краю, по центру и по нижнему краю. Для изменения вертикального выравнивания используется свойство *Вертикальное выравнивание* компонента *Текст*. Для управления выравниванием по вертикали рекомендуется использовать кнопки на панели инструментов Форматирование.

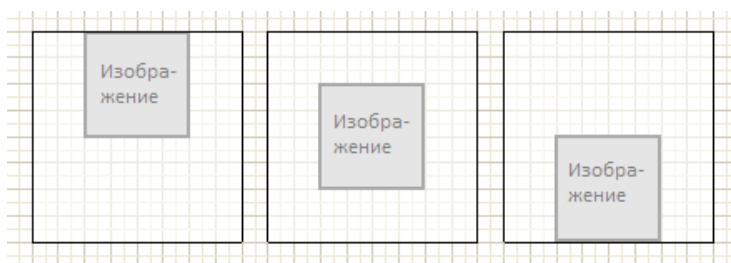


- *По верхнему краю* - содержимое компонента выравнивается относительно верхней границы компонента. После нажатия устанавливает всем выделенным компонентам выравнивание по вертикали по верхней границе компонента;
- *По центру* - содержимое компонента выравнивается по центру относительно верхней и нижней границы компонента. После нажатия устанавливает всем выделенным компонентам выравнивание по центру относительно верхней и нижней границы компонента;
- *По нижнему краю* - содержимое компонента выравнивается относительно нижней границы компонента. После нажатия устанавливает всем выделенным компонентам выравнивание по вертикали по нижней границе компонента.

Вертикальное выравнивание текста. По умолчанию текст выравнивается относительно верхней стороны. Но если возникает необходимость можно установить необходимое выравнивание. При этом если установлено выравнивание по нижней стороне и текст не вмещается по вертикали в границах компонента, он будет обрезан по верхней стороне. Если установлено выравнивание по центру, то в случае если текст не умещается, он будет обрезан одновременно по верхней и по нижней стороне.



Вертикальное выравнивание изображения. Для управления выравниванием по вертикали для компонента *Картинка* используются те же самые элементы управления, что и для компонента *Текст*. Изображения выравниваются только в случае если свойство *Растянуть* установлено в *Нет*. Иначе параметры выравнивания будут игнорироваться.

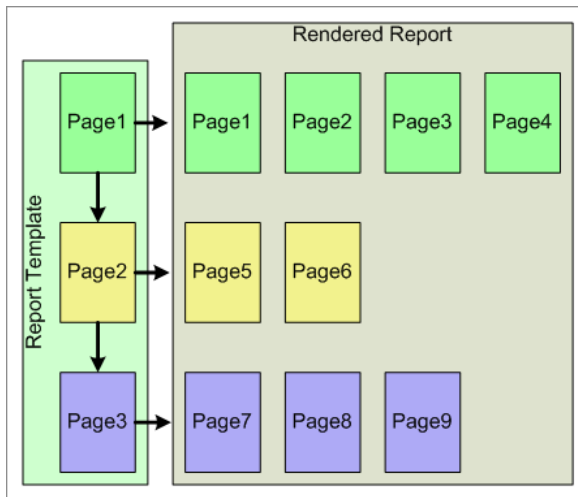


Создание отчета

В генераторе отчетов шаблон отчета делится на страницы. Каждая страница может иметь свои размеры и поля печати. Все компоненты в отчете располагаются на страницах. При построении отчета генератор отчетов последовательно обрабатывает все страницы отчета.

Порядок построения отчета

В отличие от большинства других генераторов отчетов, в генераторе отчетов шаблон отчета делится на страницы. Каждая страница может иметь свои размеры и поля печати. Все компоненты в отчете располагаются на страницах. При построении отчета генератор отчетов последовательно обрабатывает все страницы отчета.

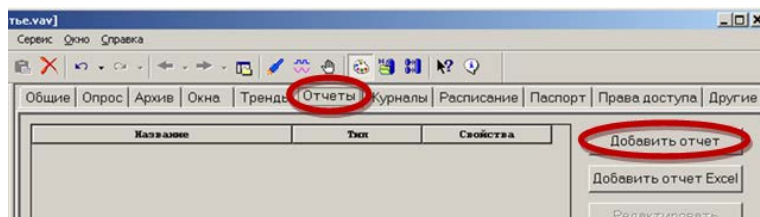


Такая структура отчета дает большую гибкость при построении отчета. При выводе страниц на печать можно отключать некоторые страницы, или наоборот включать. Можно менять порядок включения страниц в отчет. Между страницами можно организовывать взаимосвязи. При использовании компонента *Вложенный отчет* нет необходимости ссылаться на внешние отчеты, т.к. *Вложенным отчетом* в генераторе отчетов также является одна из страниц отчета.

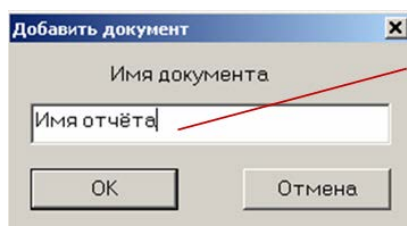
Создание нового отчёта

Для создания нового отчёта необходимо:

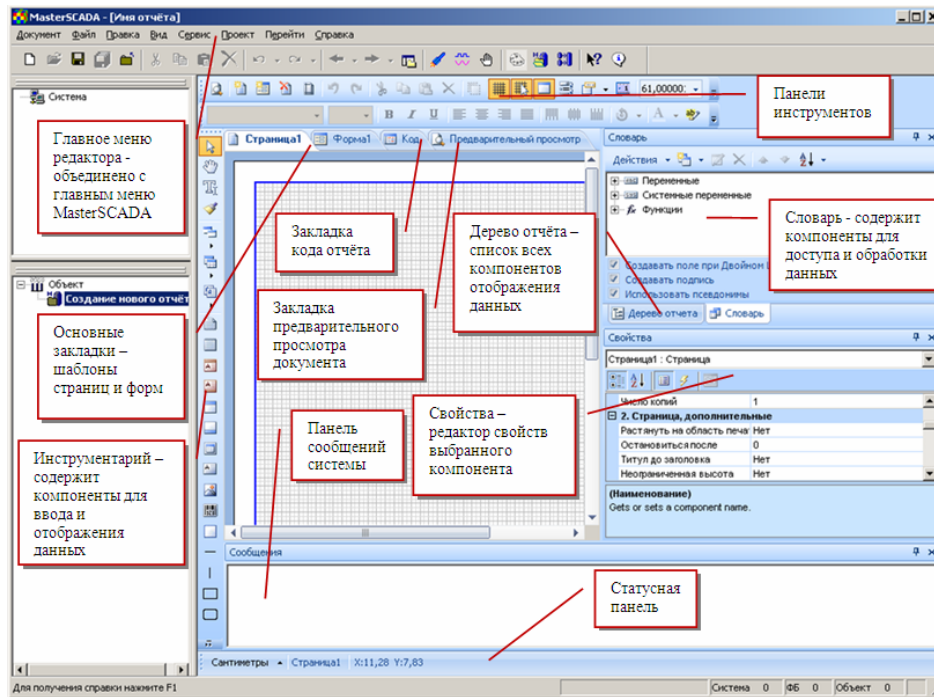
- Перейти на закладку *Отчеты* у объекта и нажать кнопку *Добавить отчет*.



- Откроется окно, в котором нужно указать имя отчета.



- Откроется редактор отчётов. Рабочая область редактора отчётов состоит из нескольких компонентов.



Обратите внимание, что все панели можно настраивать, перемещать, группировать, сворачивать, прикреплять к краям и отключать. Все компоненты генератора отчёта можно разделить на две группы:

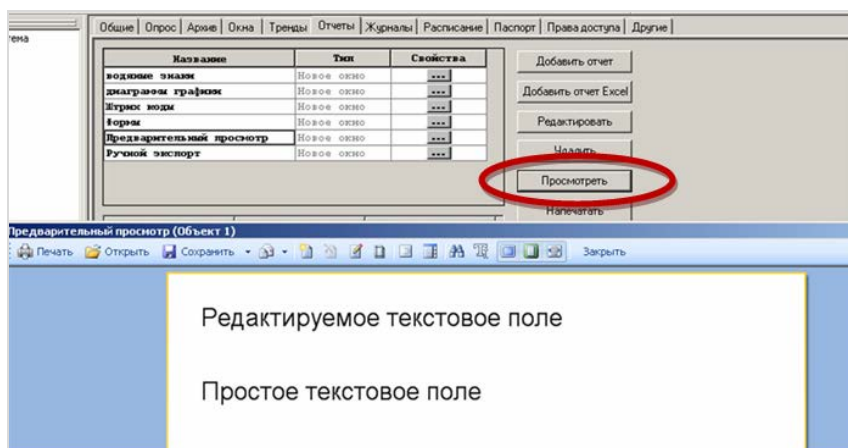
- Компоненты для отображения данных, например, компонент *Текст*;
- Компоненты для обработки данных, например, компонент *Переменная*.

Построение документа состоит из 2 этапов:

- *Создание шаблона отчёта*. Шаблон отчёта – это проект документа, локальные переменные которого не связаны с переменными MasterSCADA;
- *Создание отчёта на основе шаблона* (нужно привязать локальные переменные отчёта к переменным MasterSCADA).

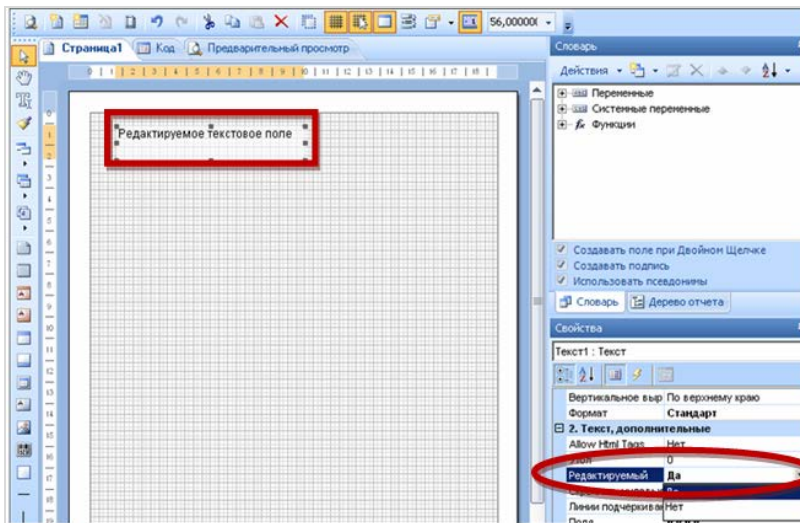
Предварительный просмотр и правка отчета

В процессе создания или редактирования отчета перейти в режим предварительного просмотра довольно просто: нужно лишь щелкнуть на закладке *Предварительный просмотр*. Также существует возможность перехода в режим предварительного просмотра в момент разработки или исполнения проекта, но в таком случае предварительный просмотр отчета будет осуществлен в отдельном окне. При этом весь инструментарий предварительного просмотра будет присутствовать. Для перехода в режим предварительного просмотра отчета в режиме разработки проекта необходимо перейти на закладку *Отчеты* объекта MasterSCADA, выбрать нужный отчет и нажать кнопку *Просмотреть*.

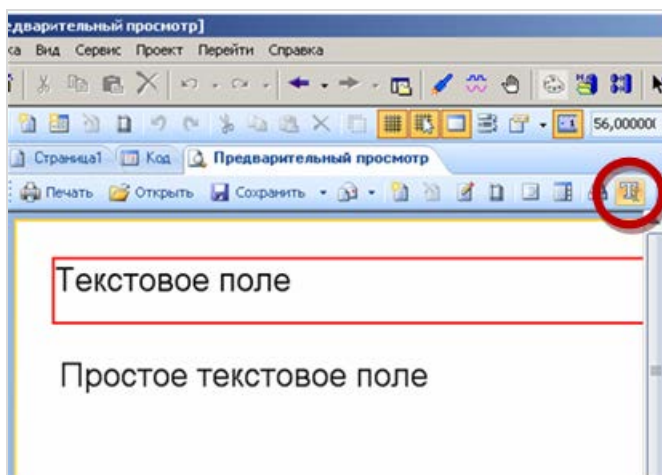


Иногда бывает необходимо не просто просмотреть полученный результат, но и отредактировать его в режиме предварительного просмотра. При этом существуют две возможности:

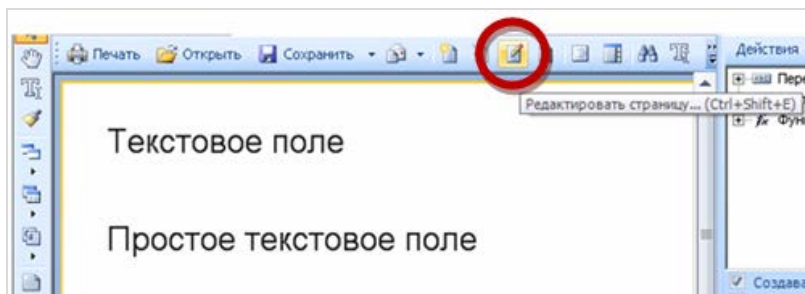
отредактировать только содержимое текстовых полей, либо отредактировать всю страницу целиком. Чтобы иметь возможность отредактировать только содержимое поля, необходимо выделить нужное поле на странице отчета и перейти к его свойствам. Далее для свойства *Редактируемый* установить значение *Да*.



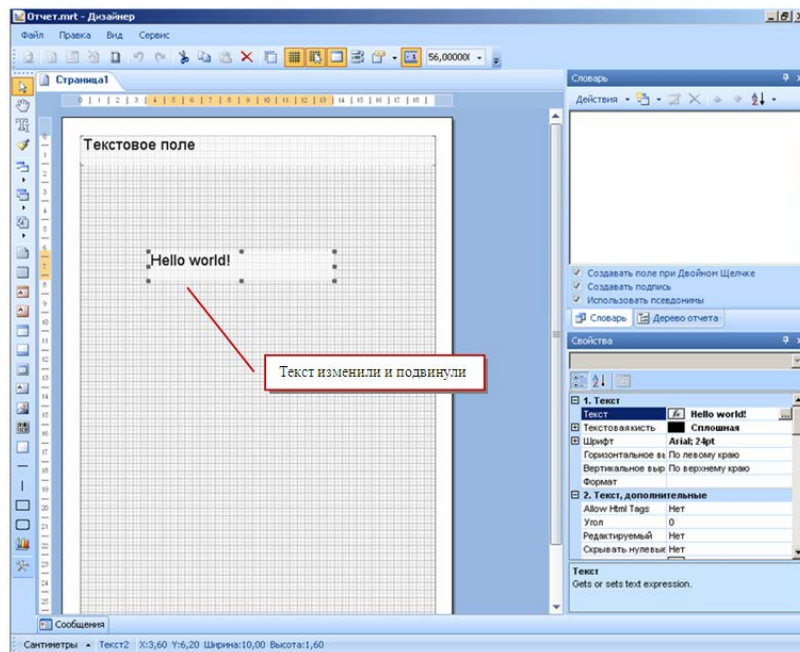
Затем перейдем в режим предварительного просмотра и щелкнем на кнопке *Редактор* панели инструментов. Текстовые поля, которые доступны для редактирования в режиме просмотра, будут автоматически выделены.



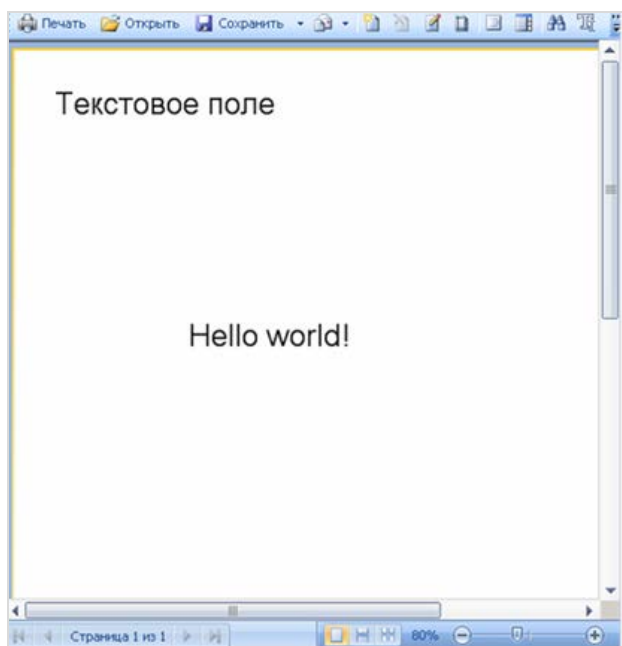
Чтобы иметь возможность отредактировать всю страницу целиком, необходимо в режиме предварительного просмотра щелкнуть на кнопке *Редактировать страницу* панели инструментов либо воспользоваться комбинацией клавиш *Ctrl+Shift+E*.



Данная страница откроется в отдельном окне в режиме дизайнера.

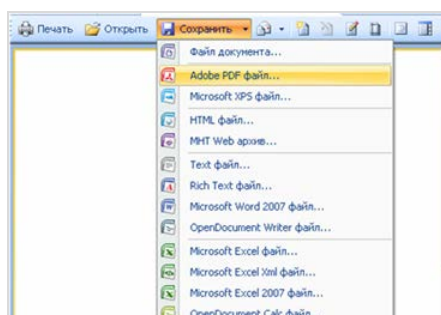


Совершив и сохранив изменения, можно оценить полученный результат, вернувшись в режим предварительного просмотра.

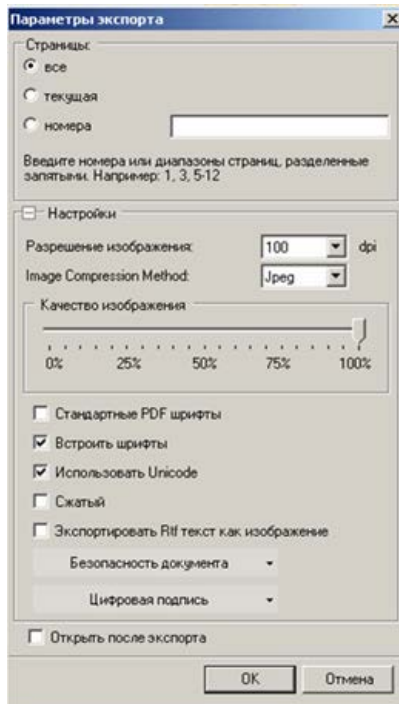


Экспорт отчета

Существует два способа экспорта отчетов: ручной и автоматический. *Ручной способ* осуществляется в режиме предварительного просмотра отчета, для этого необходимо щелкнуть на кнопке *Сохранить* панели инструментов и в выпадающем меню выбрать необходимый формат конечного файла.

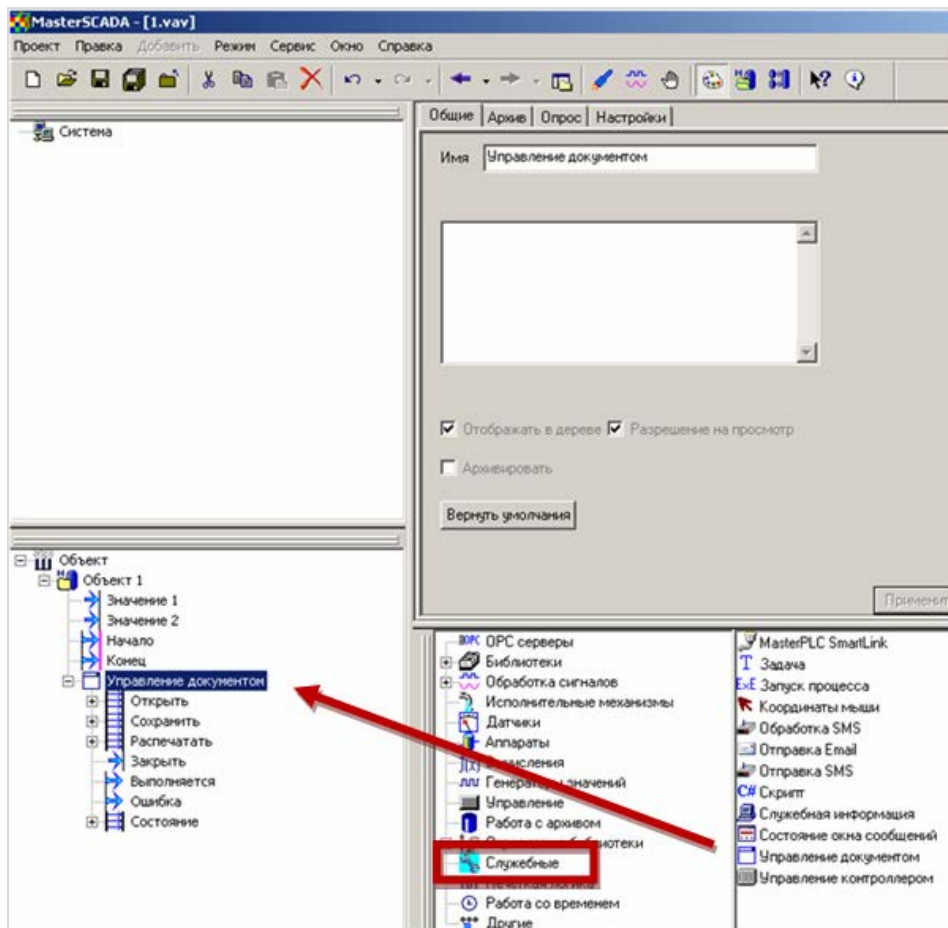


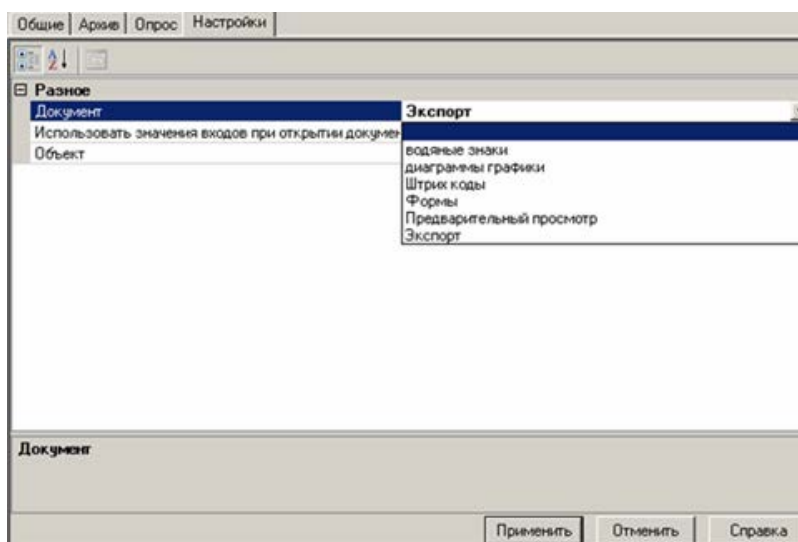
Далее в появившемся окне *Параметры экспорта* нужно указать, какие именно страницы отчета должны быть включены в экспортируемый файл, а также настроить качество изображения, безопасность документа и многое другое.



Для осуществления автоматического экспорта отчета есть три варианта:

- Через закладку *Действия* переменной *Команда* или *Событие* проекта MasterSCADA;
- Через действия на закладке *Расписание* любого объекта MasterSCADA;
- С помощью ФБ *Управление документом*. Для этого надо перетащить данный ФБ из группы *Служебные библиотеки* ФБ в дерево объектов MasterSCADA.



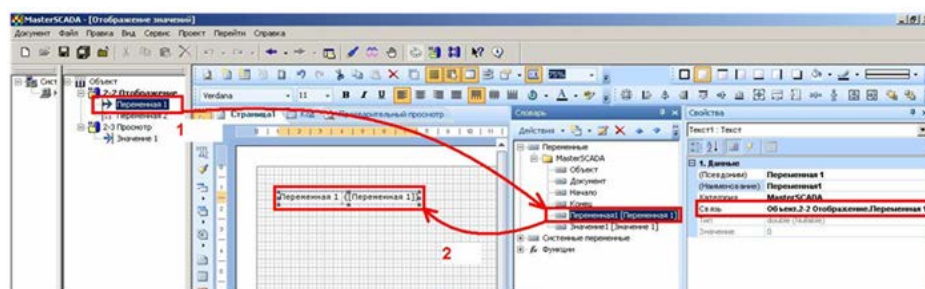


Экспорт отчета в режиме исполнения произойдет в момент появления сигнала на входа *Сохранить*. Имя сохраняемого файла ФБ возьмет со входа *Имя файла*. Тип экспорта определяется, исходя из расширения файла в полученном имени.

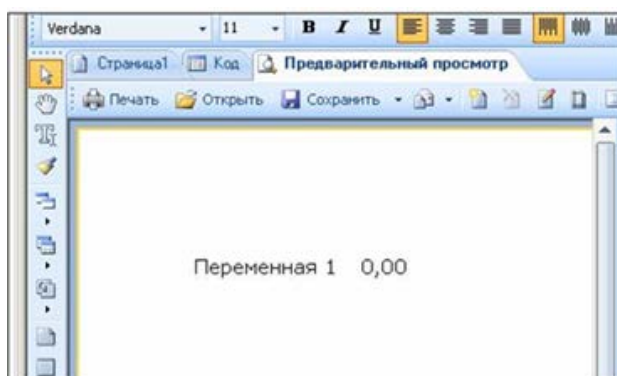
Отображение текущего значения переменной

При построении отчета можно использовать переменные, хранящие текущее значение переменных MasterSCADA или создать внутренние - локальные переменные.

Добавление переменных из дерева объектов. Чтобы отобразить значение переменной (или группы переменных) в отчёте достаточно перетащить её из дерева MasterSCADA на закладку *Словарь* редактора отчётов, а затем из словаря – на страницу отчёта. При этом в ветви переменные появятся её имя, а на закладке *Свойства* отобразится абсолютный путь этой переменной в дереве MasterSCADA. На странице отчёта вновь добавленная переменная отображается в виде двух полей: текстовым полем с именем переменной и собственно полем мгновенного значения переменной.



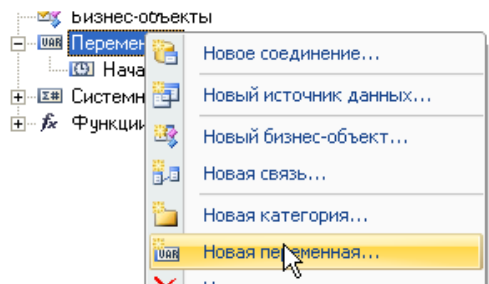
При этом, если у соответствующей переменной в дереве MasterSCADA установлено свойство *Значение до опроса* или *Константа*, то это значение можно увидеть в отчёте сразу же, выбрав закладку *Предварительный просмотр*. На этой закладке удобно контролировать все действия по изменению содержимого и оформления отчёта. Вид отчета будет полностью сформирован без перехода в режим исполнения.



- ❖ **Важно!** В отчете допускается использование переменных разных объектов MasterSCADA, но правильнее использовать только те переменные, которые принадлежат тому же объекту (или его дочерним объектам), что и отчет. Это избавит в дальнейшем от необходимости помнить и восстанавливать все внешние связи при тиражировании объекта с отчетом. Также важно помнить, что переменные добавленные из дерева объектов имеют атрибут *Только чтение*, то есть их нельзя изменить в процессе построения отчета.

Создание локальных переменных. В редакторе отчетов также можно создать локальные переменные, которые можно использовать для внутренних вычислений или для обработки переменных из дерева объектов.

Переменная добавляется через контекстное меню:



В появившемся окне можно задать имя переменной, ее тип, и способ инициализации.

При способе инициализации *Выражение* есть возможность работать с другими переменными, то есть можно инициализировать переменную результатом расчета над другой переменной. Например, можно прибавить в коде к переменной времени 1 месяц и получить дату конца отчета.

- ❖ **Важно!** Если переменная в коде инициализации обращается к другим переменным, то у нее необходимо установить флаг *Только чтение*. Изменить эту переменную в процессе построения отчета нельзя.

Если требуется изменять значение переменной в процессе построения отчета через события, то используйте способ инициализации *Значение*, а затем присваивайте переменной значение, используя событие *Начало построение отчета*.

Способы формирования отчетов

Существует три способа формирования отчетов по архивным данным значений переменной MasterSCADA: *по изменению*, *периодический* и *итоговый*.

Формирование отчетов по периодической выборке значений. Одним из способов формирования отчета является периодический. Для его задания необходимо для свойства *Способ формирования значений* источника данных выбрать *Периодический*. При таком способе строки таблицы значений будут сформированы по заданным интервалам времени. При этом на каждый конкретный интервал времени может приходиться от нуля до нескольких

значений переменной, поэтому необходимо задать способ вычисления значения. С этой целью нужно выбрать в дереве словаря колонку, перейти к ее свойству *Обработка* и присвоить ему одно из значений: *Последнее значение* (будет выбрано последнее из имеющихся значений), *Первое значение* (будет выбрано первое из имеющихся значений), *Интеграл*, *Среднее*, *Интегральное среднее*, *Максимум*, *Минимум* и *Интерполяция*.

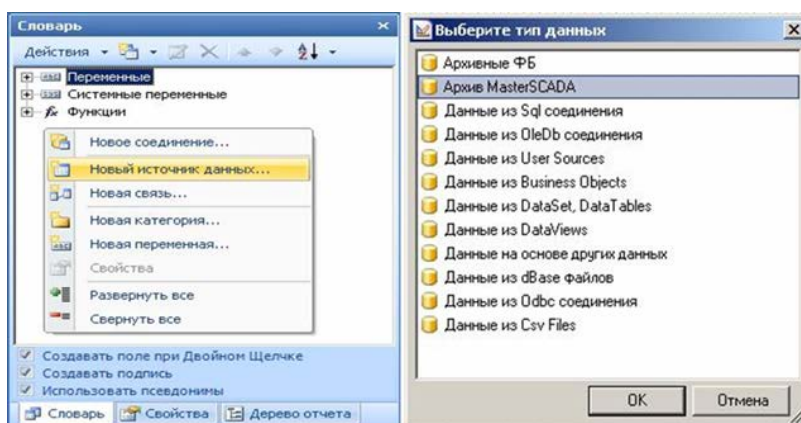
Для дискретных переменных помимо стандартных способов обработки (*Последнее значение*, *Минимум*, *Максимум* и т.д.) используются способы обработки, как *Счетчик пробега*, *Количество включений*. Счетчик пробега - определяет время наработки данной переменной, т.е. количество секунд которые переменная была во включенном состоянии, на данном интервале времени. Количество включений – определяет количество переходов из *Ложь* в *Истина* переменной на интервале времени.

Отображение архивных значений переменных

Несколько иначе, чем с мгновенными значениями, осуществляется добавление в отчет переменных, содержащих архивы значений. Для вывода архивных значений переменных MasterSCADA используется *Источник данных*.

Чтобы добавить в отчет переменную MasterSCADA, содержащую архив значений, нужно:

- Создать новый источник данных. Для этого надо щёлкнуть правой кнопкой мыши на поле панели *Словарь* и выбрать в появившемся меню пункт *Новый источник данных*, далее *Архив MasterSCADA* и присвоить ему наименование и псевдоним. Имена элементов словаря не должны содержать пробелы и специальные символы. Так же они не могут начинаться с цифры;



- Перетащить мышью в созданный источник данных переменную или группу переменных, содержащую архивные значения;
- Установить период времени, за который будут отображены архивные значения переменных. Для того, чтобы вывести данные за какой-либо период, необходимо настроить параметры источника данных *Начало* и *Конец*. Их можно задать с помощью указания определённой даты и времени, посредством вычисления функции, перетаскиванием переменной Master-SCADA. Текущую связь переменной можно посмотреть в свойствах этой переменной. Если никаких настроек не будет произведено, в отчет попадут все архивные значения переменной. Перетащить на страницу источник данных. При этом активизируется мастер добавления данных в отчет;
- В появившемся окне мастера можно:
 - Выбрать, какие из переменных источника данных будут добавлены в виде колонок таблицы в отчет, или выбрать все переменные одним щелчком;
 - Задать порядок отображения добавляемых колонок;
 - Разрешить мастеру автоматически добавить в отчет секции заголовка и итога данных;
- На вкладке *Предварительный просмотр* можно проконтролировать корректность и расположение добавленных в отчет данных.

На закладке *Свойства* источника данных можно настроить формирование строк в отчете и их поведение. *Преобразовывать Nulls* - определяет, нужно ли преобразовывать пустые значения (*Nulls*) источника данных в нули. По умолчанию используется режим *Наследовать из отчета* в этом случае настройка наследуется самого отчета. *Конвертировать в Null* недостоверные значения - если колонка имеет недостоверное значение, то при включении дан-

ной настройки результат колонки будет преобразован в *Null* (пустое значение). *Способ формирования* - определяет принцип формирования строк. Существует три способа формирования:

- *По изменению* – в этом случае каждой изменению каждой переменной, у которых включена настройка «основной», формирует новую строку;
- *Периодический* – в этом случае строки формируются с заданным у параметра *Интервал* временем;
- *Итоговый* - является частным случаем периодического. Когда *Интервал* равен единице, это интервал от начала до конца формирования отчета.

Предпочитаемый слой - если данные в MasterSCADA архивируются в базу данных, то существует возможность использовать *Слой данных*. В редакторе отчетов, также можно выбрать предпочитаемый слой, из которого будут получены данные. *Использовать другие слои* - при использовании прореженных слоев (часовых, суточных), может возникнуть ситуация, что данных на необходимый момент времени нет. Настройка определяет брать ли в этом случае редактору отчетов данные из другого слоя.

Свойства переменных источника данных

Настройка **Основной** -присутствует, если способ формирования установлен *По изменению*. Если данная настройка включена, то каждое изменение этой переменной приводит к созданию строки данных в отчете.

Настройка **Атрибут данных** - определяет, какой из атрибутов переменной выводится в отчет. Каждая переменная источника имеет три атрибута:

- *Значение* – значение переменной в архиве;
- *Время* – метка времени переменной в архиве. В случае выбора данного атрибута становится доступной настройка *Время в формате UTC*, которая определяет возвращать метку времени в глобальном времени UTC (время по Гринвичу) или в локальном времени;
- *Признак качества* – признак качества переменной в архиве. Признак качества возвращается в строковом формате (*Норма, Ошибка датчика, Ошибка устройства* и т.д.)

Настройка **Граница интервала** определяет, необходимо ли учитывать граничные значения интервала.

Настройка Обработка

При построении отчета на каждый конкретный интервал времени может приходиться от нуля до нескольких значений переменной, поэтому необходимо задать способ вычисления результата – какое из значений должна возвращать переменная источника (колонка данных). Для этого предназначена настройка **Обработка**, с помощью которой определяется способ обработки параметра.

Доступны следующие способы обработки:

- *Последнее значение* - будет выбрано последнее из значений на интервале;
- *Первое значение* - будет выбрано первое из значений на интервале;
- *Среднее* – вычисляется среднее арифметическое значений в архиве на интервале;
- *Максимум* – колонка возвращает максимальное из всех значений на интервале;
- *Минимум* – колонка возвращает минимальное из всех значений на интервале;
- *Интегральное среднее* – вычисляется среднее интегральное всех значений в архиве;
- *Интеграл* - имеет ряд дополнительных настроек по сравнению с интегральное среднее.

Для дискретных переменных помимо стандартных способов обработки (*Последнее значение, Минимум, и т.д.*) доступны также способы обработки:

- *Счетчик пробега* - определяет время наработки данной переменной, т.е. количество секунд которые переменная была во включенном состоянии, на данном интервале времени;

- *Количество включений* – определяет количество переходов из *Ложь* в *Истина* переменной на интервале времени.

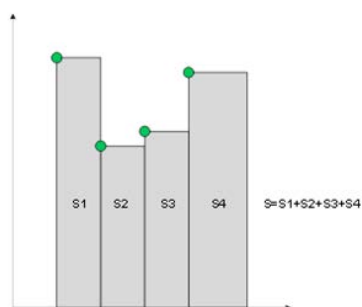
Способ обработки **Интеграл** предназначен для вычисления площади фигуры ограниченной графиком из значений архива. Данный способ обработки применяется для вычисления суммарного расхода, по его действующему значению (например, вычисление расхода газа за час по действующему значению расхода).

При выборе данного способа обработки становятся доступны следующие настройки:

- *Параметр обработки* - если в данном поле указано значение отличное от нуля, то результат вычисления интеграла делится на это значение. Данная настройка предназначена для приведения значения к реальным единицам. Например, если расходер возвращает значения расхода в м3/ч, то нужно указать в данном поле 3600, тогда колонка будет возвращать реальное значение расхода за интервал времени;
- *Метод интегрирования* – данная настройка определяет способ вычисления интеграла.

У *Метода интегрирования* доступны две настройки:

- *Метод левых прямоугольников* – в этом случае интеграл считается как сумма прямоугольников, образованных точками значений в архиве.



- *Метод трапеций* – в этом случае, между точками в архиве проводится линия. Интеграл считается как сумма образовавшихся трапеций.



Способ обработки **Интегральное среднее** предназначен для вычисления среднего интегрального значения за интервал времени. При выборе данного способа обработки происходит вычисление интеграла, а затем результат делится на длительность интервала в секундах. Интегральное среднее имеет такие же настройки что и обработка Интеграл.

Обработка плохих признаков качества

Настройка *Настройка признака результата* задает, какой признак качества будет выставляться у результата, если на интервале находятся некорректные значения влияющие на результат. Доступны 4 настройки:

- *Good* – результату устанавливается признак качества *норма*;
- *Bad* – результату устанавливается признак качества *ошибка*;
- *Uncertain* – результату устанавливается признак качества *не определено*;
- *Наследовать* – в этом случае наследуется признак результата обработки.

Если на интервале обработки будет обнаружено значение влияющее на результат с данным признаком качества, то в результате колонка будет иметь признак определенный данной настройкой. То есть, если в разделе *Обработка признаков Bad*, настройка признака результата будет задана *Uncertain*, то при обнаружении на интервале значения с признаком каче-

ства *Bad*, результату колонки будет установлен признак *Uncertain*. Алгоритм определения признака результата отличается в зависимости от способа обработки параметра:

- Если используются способы обработки *Первое значение*, *Последнее значение*, *Максимум*, *Минимум*, то признак качества колонки выставляется на основании результата. Например, если способ обработки - *Последнее значение*, последнее значение на интервале имело признак, относящийся к *Bad* (*Ошибка датчика*, *Ошибка устройства* и т.д.), то признак результата определится настройкой группы настроек *Bad*. Если на интервале есть значение с признаком *Bad*, но оно не является последним, то на результат оно не влияет;
- Если используются способы обработки *Среднее*, *Интеграл*, *Интегральное среднее* то признак качества колонки выставляется на основании всех значений на интервале - поскольку все значения влияют на результат. Например, если способ обработки *Среднее*, и на интервале было обнаружено значение с признаком качества *Bad*, то признак результата определится настройкой группы настроек *Bad*.

Если на интервале будут обнаружены значения с разными недостоверными признаками, то результату присваивается наиболее недостоверное (группа *Bad* менее достоверно, чем группа *Uncertain*).

При включении настройки **Игнорировать значения** значения с недостоверным признаком (то есть отличным от *Good* – 192), игнорируются при обработке интервала. Например, для обработки *Максимум* данная настройка определяет какое из максимальных значений выдать - найденное недостоверное или максимальное среди достоверных.

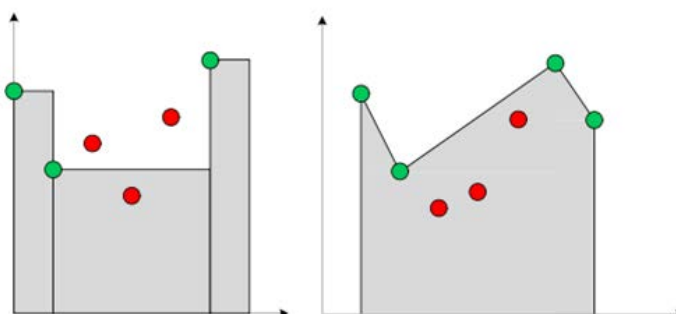
Раздел настроек **Обработка плохих признаков качества** позволяет задать различные способы обработки значений с плохими признаками качества. Настройку обработки плохих признаков можно задать независимо для трех типов некачественных значений:

- *Stop* – признак остановки SCADA системы;
- *Bad* – признак качества *Ошибка*. В данный признак качества входят ошибки. В этот признак качества входят признаки:
 - Значение недостоверно (*Bad* -0);
 - *Ошибка конфигурации* (*Config Error* – 4);
 - *Нет соединения* (*Not Connected* – 8),
 - *Ошибка устройства* (*Device Failure* - 12);
 - *Ошибка датчика* (*Sensor Failure* - 16),
 - *Последнее значение* (*Last Know* – 20);
 - *Нет связи* (*Comm Failure* – 24);
 - *Не обсуживается* (*Out of service* – 28).
 - *Uncertain* – признак качества *Не определено*. В этот признак входят признаки:
 - *Не определено* (*Uncertain* – 64);
 - Датчик не откалиброван (*Sensor Cal* – 80);
 - *Значение не допустимо* (*Egu Exceeded* – 84 и *SubNormal* - 88).

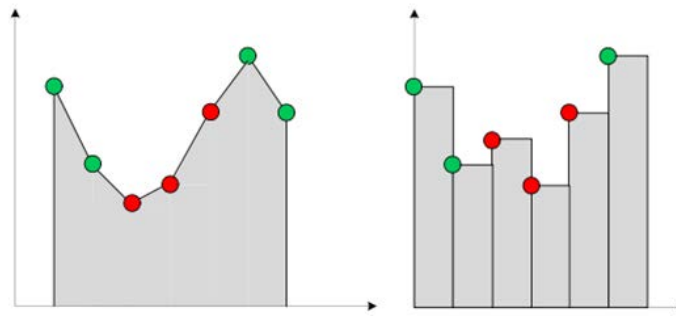
Обработка *Интеграл* имеет особенности при обработке плохих признаков качества. При включении данной обработки появляется дополнительное поле – *Настройка*. В данном поле можно задать, как обрабатывать плохие признаки качества.

Доступны четыре режима:

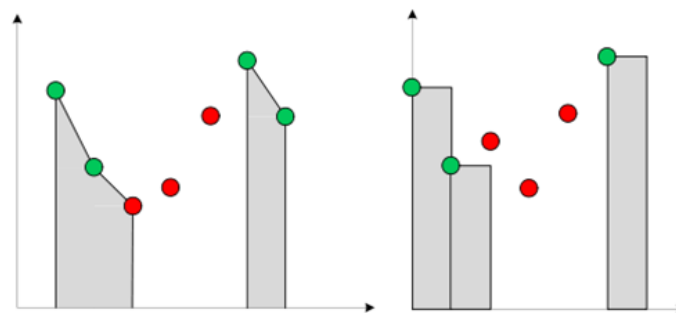
- *Игнорировать* – в этом случае интервал с плохими признаками вычисляется, на основе хороших значений. Если используется метод левых прямоугольников, то интервал вычисляется на основе последнего хорошего значения, а если метод трапеций – то между хорошими значениями проводится линия, и вычисляется площадь получившийся трапеции;



- *Игнорировать признак* – в этом режиме, вычисляется площадь всех интервалов независимо от признака качества;



- *Исключать недостоверные области* – в этом случае вычисляется площадь интервала, содержащем хорошее и плохое значение, от последнего хорошего, до первого плохого значения. Площади интервалов с остальными плохими значениями не вычисляются;



- *Настройка* – позволяет произвести гибкую настройку способа интегрирования. При включении данного режима, становятся активными настройки *Режим интегрирования* и *Игнорировать значения*.

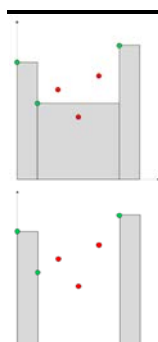
Режим интегрирования - определяет способ интегрирования плохих значений. Доступны следующие настройки - *Интегрировать до*, *Интегрировать после*, *Интегрировать весь*, *Не интегрировать*. На интервале времени может возникнуть ситуация когда на нем присутствуют значения с разными признаками недостоверности – например, значения категории *Bad* и *Uncertain*, имеющие разные настройки. Общий алгоритм вычисления результата интегрирования следующий:

- Если две соседние точки *хорошие*, то интеграл считается по этим значениям;
- Если одна точка *плохая*, другая *хорошая*, то учитывается настройка для плохого признака;
- Если обе *плохие*, то область будет интегрироваться только, если настройки обеих точек указывают, что ее необходимо считать.

Далее приведены все возможные комбинации настроек *Метод интегрирования*, *Режим интегрирования* (плохих значений) и *Игнорировать значения*. Зелеными точками обозначены значения в архиве имеющие хороший признак качества, красными – плохой.

Способ вычисления – метод левого прямоугольника.

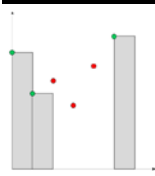
Способ вычисления – метод левого прямоугольника.



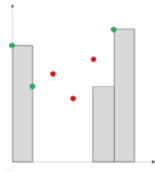
Левый прямоугольник – Игнорировать – Интегрировать весь. В этом режиме, площадь интервала с плохими значениями считается по последнему корректному значению.

Левый прямоугольник – Игнорировать – Не интегрировать. В этом режиме, площадь интервала с плохими значениями не вычисляется.

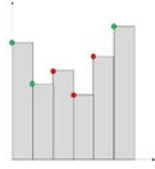
Способ вычисления – метод левого прямоугольника.



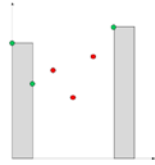
Левый прямоугольник – Игнорировать – Интегрировать до. В этом режиме, вычисляется площадь интервала от последнего хорошего, до первого плохого значения. Площади интервалов с остальными плохими значениями не вычисляются.



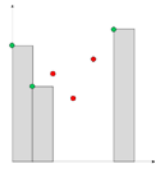
Левый прямоугольник – Игнорировать – Интегрировать после. В этом режиме, вычисляется площадь интервала от последнего плохого, до первого хорошего значения (по значению последнего хорошего значения). Площади интервалов с остальными плохими значениями не вычисляются.



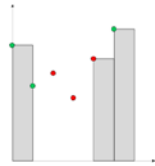
Левый прямоугольник – Не игнорировать – Интегрировать весь. В этом режиме, вычисляется площадь всех интервалов независимо от признака качества.



Левый прямоугольник – Не игнорировать – Не интегрировать. В этом режиме, площадь интервала с плохими значениями не вычисляется.

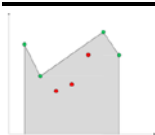


Левый прямоугольник – Не игнорировать – Интегрировать до. В этом режиме, вычисляется площадь интервала от последнего хорошего, до первого плохого значения. Площади интервалов с остальными плохими значениями не вычисляются.



Левый прямоугольник – Не игнорировать – Интегрировать после. В этом режиме, вычисляется площадь интервала от последнего плохого, до первого хорошего значения (по значению последнего плохого значения). Площади интервалов с остальными плохими значениями не вычисляются.

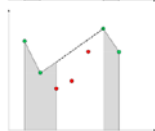
Способ вычисления – метод трапеций.



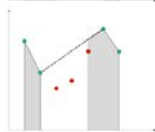
Трапеция – Игнорировать – Интегрировать весь. В этом режиме, между последним хорошим значением, и первым хорошим значения проводится прямая линия. Затем вычисляется площадь сформированной трапеции



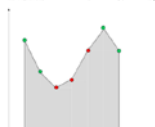
Трапеция – Игнорировать – Не интегрировать. В этом режиме, интервалы с плохими значениями не вычисляются.



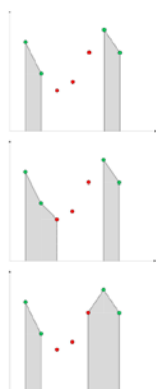
Трапеция – Игнорировать – Интегрировать до. В этом режиме, между последним хорошим значением, и первым хорошим значением проводится прямая линия. Затем вычисляется площадь сформированной трапеции ограниченная последним плохим значением.



Трапеция – Игнорировать – Интегрировать после. В этом режиме, между последним хорошим значением, и первым хорошим значением проводится прямая линия. Затем вычисляется площадь сформированной трапеции ограниченная последним плохим значением.



Трапеция – Не игнорировать – Интегрировать весь. В этом режиме, вычисляется площадь всех интервалов независимо от признака качества.



Трапеция – Не игнорировать – Не интегрировать. В этом режиме, площадь интервала с плохими значениями не вычисляется.

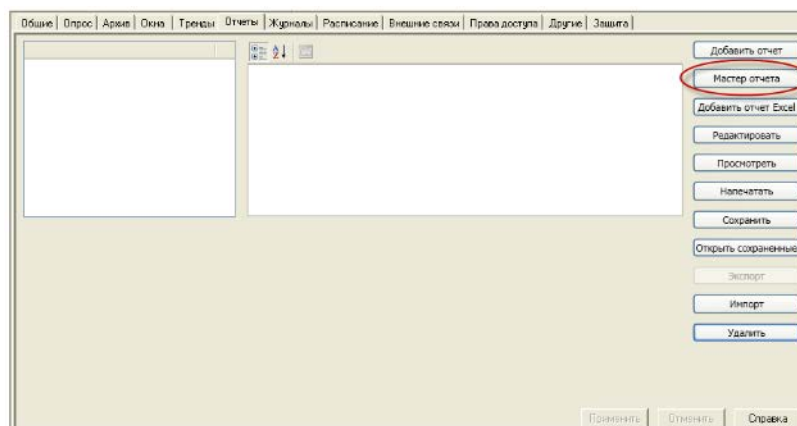
Трапеция – Не игнорировать – Интегрировать до. В этом режиме, между последним хорошим значением, и первым плохим значением проводится наклонная линия, после чего вычисляется площадь сформированной трапеции. Остальные интервалы с плохими значениями игнорируются.

Трапеция – Не игнорировать – Интегрировать после. В этом режиме, между последним плохим значением, и первым хорошим значением проводится наклонная линия, после чего вычисляется площадь сформированной трапеции. Остальные интервалы с плохими значениями игнорируются.

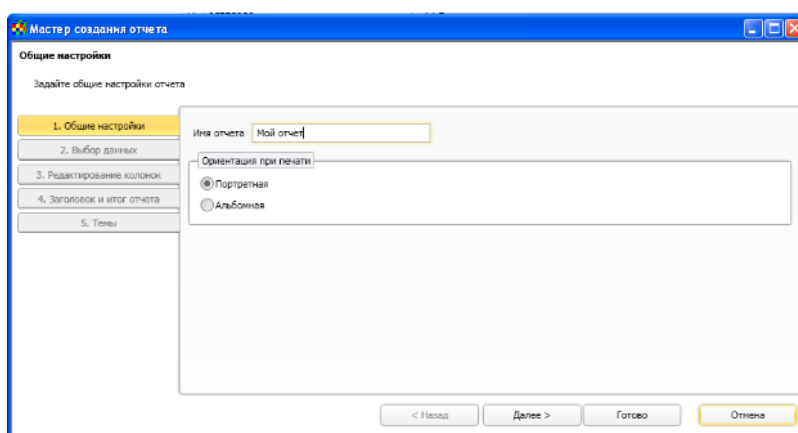
Мастер отчетов

Мастер отчетов позволяет быстро создавать шаблонные отчеты на основе собственных данных. Созданный в мастере отчет можно впоследствии редактировать как обычный отчет – добавлять источники, колонки, изменять настройки и т.д.

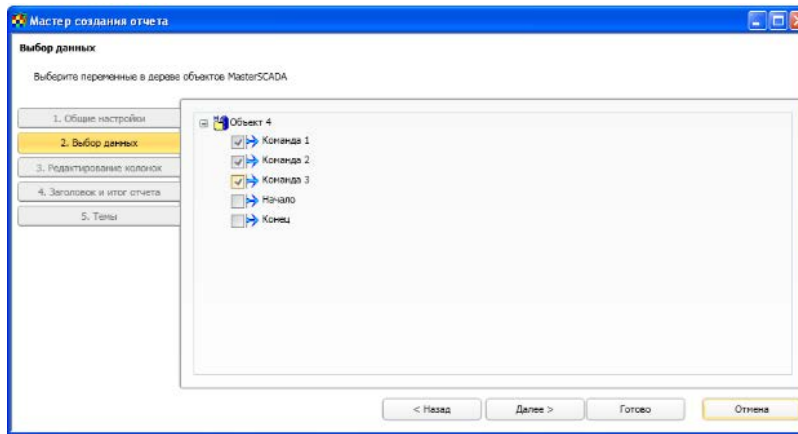
Мастер создания отчетов запускается кнопкой на закладке объекта *Отчеты*.



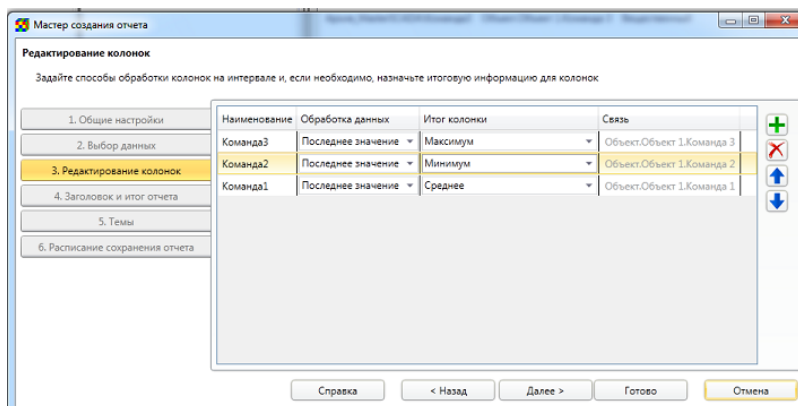
Следующим шагом нужно задать имя, которое будет иметь отчет. Также на странице можно задать ориентацию листа.



Далее осуществляется выбор переменных, которые необходимо отображать в отчете. Для этого необходимо в дереве, повторяющем структуру дерева текущего объекта, отметить галочками нужные переменные.



После этого необходимо настроить добавленные колонки – способ их обработки и способ вычисления итога. Настройка осуществляется в таблице.



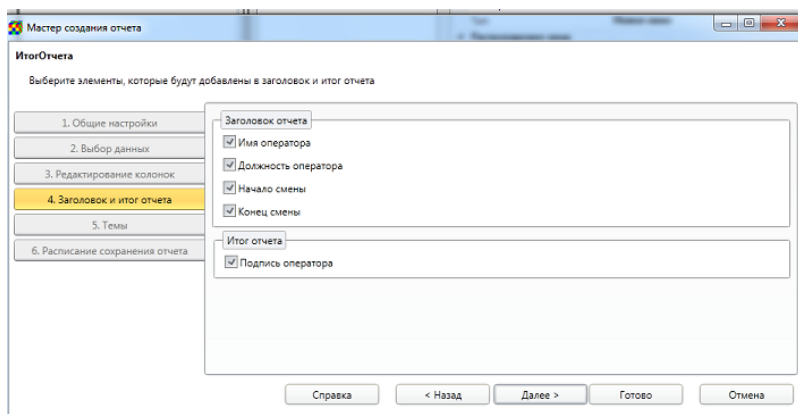
Наименование - определяет имя колонки, которое она будет иметь в отчете. **Обработка данных** - определяет способ обработки параметра, например минимум - минимальное значение на интервале, последнее - последнее значение на интервале и т.д. **Итог** - определяет метод расчета итога для каждой колонки. **Связь** - нередактируемое поле, отображающее связь с переменной дерева объектов.

После этого можно выбрать информацию, которую необходимо вывести в заголовок и итог отчета.

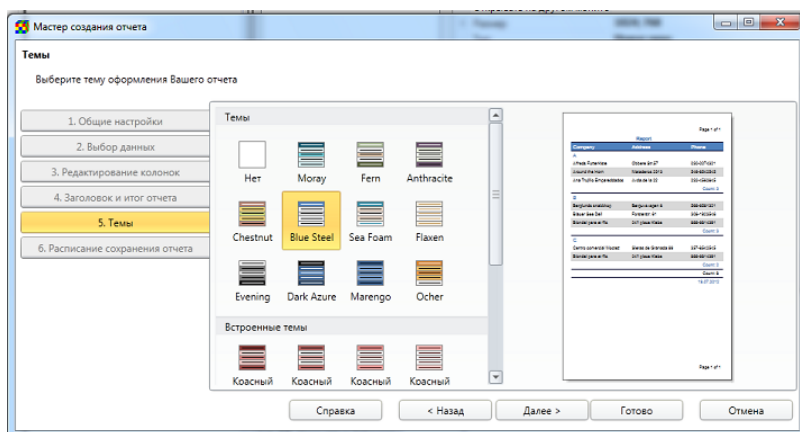
В заголовке отчета можно добавить:

- *Имя оператора* – имя текущего оператора;
- *Должность оператора* – должность текущего оператора;
- *Начало смены* – время начало смены текущего оператора;
- *Конец смены* – время конца смены текущего оператора.

В итог отчета можно добавить *Подпись оператора* – в итог отчета добавляется поле *Подпись* и горизонтальная линия для подписи.



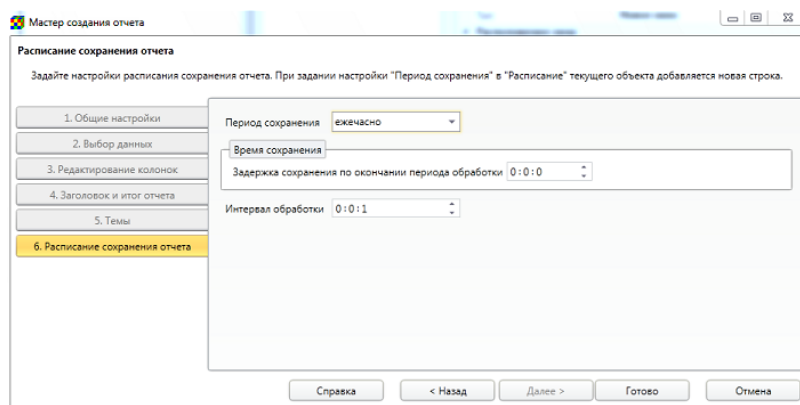
Далее необходимо задать цветовое оформление отчета.



После этого настраивается расписание сохранения отчета. Настройка задается в раскрываемом списке *Период сохранения*. Доступны следующие типы сохранения:

- *Не сохранять*. В этом случае отчет не будет сохраняться автоматически. Для корректной работы необходимо создать в дереве объектов переменные определяющие начало и конец отчета (диапазон выборки из архива) и связать их с параметрами *Начало* и *Конец* источник данных *Архив MasterSCADA*;
- *Ежечасно*. Отчет будет сохраняться каждый час. В настройках можно задать задержку, по прошествии которой начинается сохранения отчета;
- *Ежедневно*. Отчет будет сохраняться ежедневно. В настройках можно задать задержку, по прошествии которой начинается сохранения отчета;
- *Еженедельно*. Отчет будет сохраняться в заданный день недели. В настройках можно задать задержку, по прошествии которой начинается сохранения отчета;
- *Ежемесячно*. Отчет будет сохраняться в заданный день месяца. В настройках можно задать задержку, по прошествии которой начинается сохранения отчет

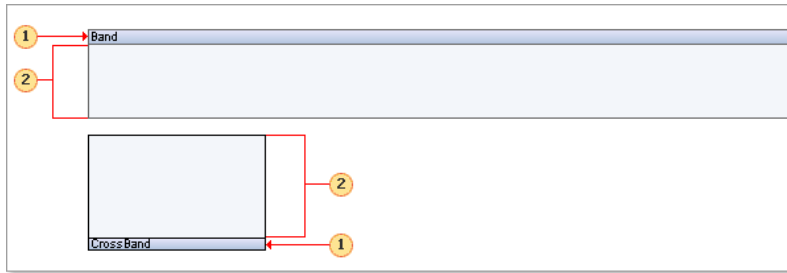
У каждого способа сохранения отчета можно задать параметр *Интервал обработки* - данный параметр определяет шаг времени вывода строк данных в отчете (в секундах), то есть с каким интервалом будут формироваться строки на странице.



После выполнения всех шагов созданный отчет можно запустить в режим предварительного просмотра, т.е. просмотреть уже построенный отчет (при наличии всех данных), либо открыть в режиме редактирования, чтобы посмотреть, как реализован созданный отчет, и при необходимости доработать его. В режиме редактирования можно вносить изменения в отчет стандартными средствами редактора отчетов.

Секции отчета

Генератор отчетов строит отчет при помощи секций (иногда секции также называют бэндами). Секция состоит из двух частей - заголовка и рабочей зоны. В заголовке отображается наименование секции. Кроме того, в заголовке может отображаться различная системная информация. Каждая секция является контейнером и может содержать другие компоненты. Позиция компонентов на секции считается относительно позиции верхнего левого угла содержимого секции. Под цифрами 1 на рисунке ниже – наименование секции, цифра 2 - рабочая зона.



Секции не имеют никакого графического представления в построенном отчете. Оформление секции достигается путем оформления компонентов расположенных на нем. Свойства секции управляют только тем, каким образом секция будет расположена в построенном отчете. Обычно отчет состоит из множества секций с текстовыми и графическими компонентами, расположенными на них. При построении отчета секции многократно копируются вместе с находящимися на них компонентами. К примеру, секция *Заголовок* выводится один раз в начале данных. А секция *Данные* выводится для каждой строки данных.

Типы секций

В генераторе отчетов есть множество различных типов секций, каждый из которых обладает специфическими возможностями. Все секции, в свою очередь, разделяются на две категории: секции и кросс секции. Кросс-секции не могут быть расположены на странице, но зато их можно расположить на простых секциях. Простые секции являются базовым элементом для построения отчетов. Ниже в таблице приведены все простые секции.

Иконка	Название	Описание
	Заголовок отчета	Печатается один раз в начале отчета
	Итог отчета	Печатается один раз в конце отчета
	Заголовок страницы	Печатается сверху на каждой странице
	Итог страницы	Печатается внизу на каждой странице
	Заголовок группы	Печатается в начале группы
	Итог группы	Печатается в конце группы
	Заголовок колонки	Печатается перед колонкой
	Итог колонки	Печатается после колонки
	Заголовок данных	Печатается перед данными
	Итог данных	Печатается после данных
	Данные	Печатается столько раз, сколько записей в источнике данных
	Дочерний	Печатается один раз сразу за бэндом, после которого он расположен

Для удобства визуального восприятия все секции имеют свой цвет. Это позволяет легко понимать структуру отчета.

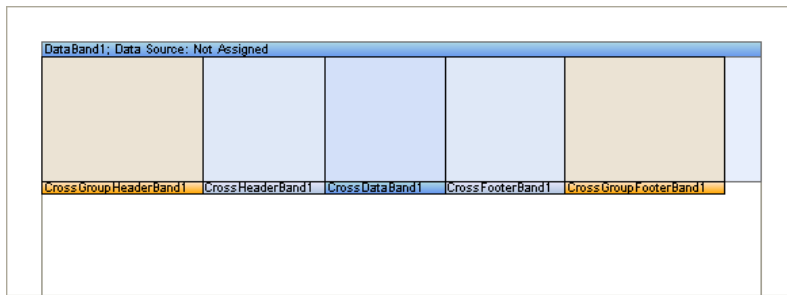


Вторая категория секций – это кросс-секции. Секции этой категории нельзя положить на страницу, но их можно положить на любую простую секцию. Это позволяет строить очень сложные кросс отчеты. Список кросс-секций приведен ниже.

❖ **Важно!** Кросс-секции могут быть расположены только на простых секциях. Их нельзя расположить, к примеру, на странице или в контейнере.

Иконка	Название	Описание
	Кросс-заголовок группы	Печатается в начале группы
	Кросс-итог группы	Печатается в конце группы
	Кросс-заголовок данных	Печатается перед данными
	Кросс-итог данных	Печатается после данных
	Кросс-данные	Печатается столько раз, сколько записей в источнике данных

В отличие от простых секций, у кросс-секций заголовков расположен не сверху секции а снизу.





Порядок вывода секций

При построении отчета все секции выводятся в строго определенном порядке. Это происходит, потому что каждая секция выполняет определенную функцию в отчете. И очень важен порядок, в котором будут выводиться на печать секции. К примеру, при выводе таблицы используются три секции: *Заголовок*, *Данные*, *Итог*. Секция *Заголовок* служит для размещения заголовков данных. Сами данные размещаются на секции *Данные*, а итоги по этим данным размещаются на секции *Итог*.

Порядок	Название	Описание
1	Заголовок страницы	На каждой странице. Вывод на первой странице – опционально.
2	Заголовок отчета	Один раз в начале отчета. <i>Заголовок отчета</i> может выводиться раньше чем <i>Заголовок страницы</i> , если у страницы (на которой находятся обе секции) свойство <i>Титул</i> до заголовка установлено в <i>Да</i> .
3	Заголовок данных, Заголовок колонки	Один раз перед выводом данных (Для <i>Заголовок колонки</i> - один раз для каждой колонки). Вывод на каждой новой странице – опционально.
4	Заголовок группы	В начале каждой группы. Вывод на каждой новой странице – опционально.
5	Данные	Для каждой строки данных.
6	Итог группы	В конце каждой группы.
7	Итог данных, Итог колонки	После вывода всех данных (Для <i>Итог колонки</i> - один раз для каждой колонки). Вывод на каждой новой странице – опционально.
8	Итог отчета	Один раз в конце отчета.
9	Итог страницы	На каждой странице. Вывод на первой странице – опционально.

В таблице не указана секция *Дочерний*. Эта секция всегда печатается после той секции, после которой она расположена на листе.

- ❖ **Важно!** В первую очередь выводятся компоненты, которые не расположены ни на одной секции. После этого выводятся секции.

В одном отчете возможно использовать достаточно большое количество секций одного типа. К примеру две секции *Заголовок*. Все секции одинакового типа выводятся в порядке расположения на странице. Т.е. сначала выведется секция, которая выше других, затем, секция которая находится под ней и т.д. В случае с кросс-секциями правило тоже самое, только первой будет выведена секция, которая находится левее остальных. Порядок расположения секций можно менять. Для этого, просто перетащите курсором мышки одну из секций выше или ниже другой или воспользуйтесь командами *Переместить вперед* и *Переместить назад*. Используйте кнопки   на панели инструментов *Дизайн* для изменения порядка компонентов.

Вывод на одну строку двух секций

Чтобы вывести на одну строку данных две секции можно использовать секцию *Дочерний*. Создадим новый отчет. На страницу положим секцию *Данные*, ниже которой поместим секцию *Дочерний*.

DataBand1; Data Source: Customers		
{Customers.CompanyName }	{Customers.Address }	{Customers.Phone }
ChildBand1		
Child		

Запустим отчет на выполнение. Как видно из примера, секция *Дочерний* была выведена на печать столько же раз сколько и секция *Данные*. Т.е. секция *Дочерний* является как бы продолжением секции *Данные*, но при этом она остается секцией со всеми присущими ей свойствами.

Alfreds Futterkiste	Obere Str. 57	030-0074321
Child		
Ana Trujillo Emparedados y helados	Avda. de la Constitución 22	(5) 555-4729
Child		
Antonio Moreno Taquería	Mataderos 2312	(5) 555-3932
Child		
Around the Horn	120 Hanover Sq.	(171) 555-7788
Child		
Berglunds snabbköp	Berguvsvägen 8	0921-12 34 65
Child		
Blauer See Delikatessen	Forsterstr. 57	0621-08460
Child		
Blondesddsl père et fils	24, place Kléber	88.60.15.31
Child		
Bólido Comidas preparadas	C/ Araquil, 67	(91) 555 22 82
Child		
Bon app'	12, rue des Bouchers	91.24.45.40
Child		
Bottom-Dollar Markets	23 Tsawassen Blvd.	(604) 555-4729
Child		
B's Beverages	Fauntleroy Circus	(171) 555-1212
Child		
Cactus Comidas para llevar	Cerrito 333	(1) 135-5555
Child		
Centro comercial Moctezuma	Sierras de Granada 9993	(5) 555-3392
Child		
Chop-suey Chinese	Hauptstr. 29	0452-076545
Child		
Comércio Mineiro	Av. dos Lusíadas, 23	(11) 555-7647

Необходимо отметить, что секцию *Дочерний* можно использовать не только с секцией *Данные*. Ее можно расположить после любой секции на странице. К примеру, после секции *Заголовок* или после секции *Заголовок группы*. Секцию *Дочерний* можно использовать совместно с любыми секциями, расположенными на странице.

Нумерация строк данных

Кроме данных и заголовков данных в отчетах еще используются и подвалы данных. В подвалах обычно выводятся различные итоги. Рассмотрим, как добавить к списку подвал. Для этого воспользуемся секцией *Итог*. Эта секция размещается после секции *Данные*. По умолчанию, секция *Итог* печатается один раз, после того как будут напечатаны все данные. Добавим на страницу секцию *Итог*, на которой разместим один текстовый компонент. Предположим, этот текстовый компонент, будет выводить количество выведенных на печать строк.

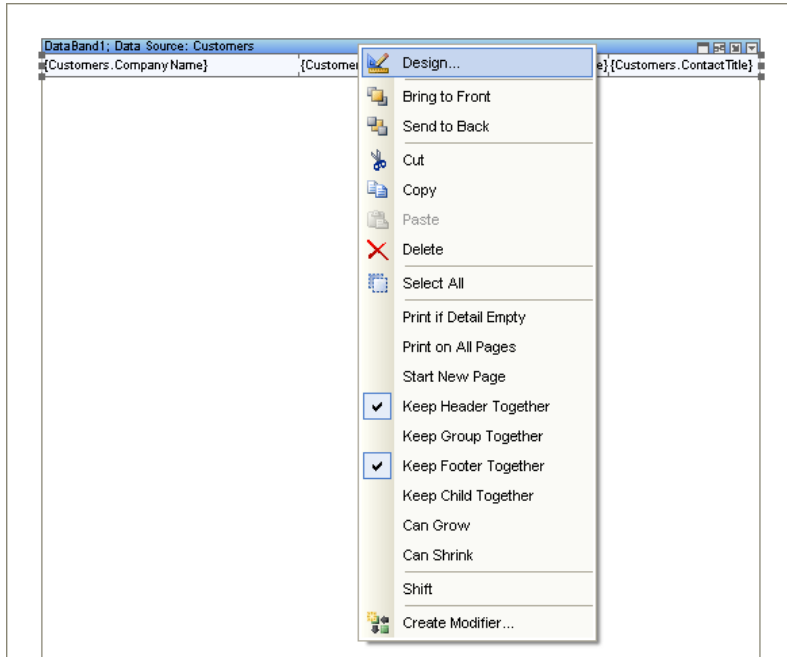
Секции данных

К секциям данных относятся три секции: *Заголовок данных*, *Итог данных* и *Данные*. При помощи этих секций выводятся данные в отчете. Основной является секция *Данные*. Каждой секции *Данные* указывается источник данных. Источник данных фактически представляет из себя таблицу. В каждом источнике данных есть поля данных. Разместив на секции данных текстовые компоненты со ссылками на эти поля, можно вывести на печать таблицу. Один источник данных может указывать на неизвестное заранее количество строк с данными.

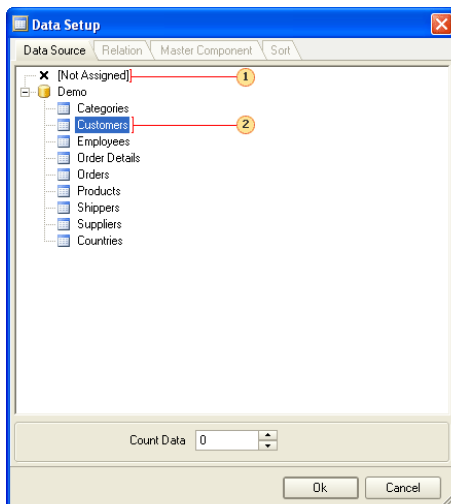
Секция *Данные* выводится столько раз, сколько строк в указанном источнике данных. К примеру, если в источнике данных есть 100 строк, то секция *Данные* будет выведена на печать ровно 100 раз. Если при выводе на печать заканчивается место на странице, то будет сформирована новая страница и печать будет продолжена. При помощи секции *Заголовок дан-*

ных в отчет добавляются заголовки к таблице, которая выводится при помощи секции *Данные*. Соответственно, секция *Итог данных* используется для того чтобы вывести различные итоги по выводимой таблице.

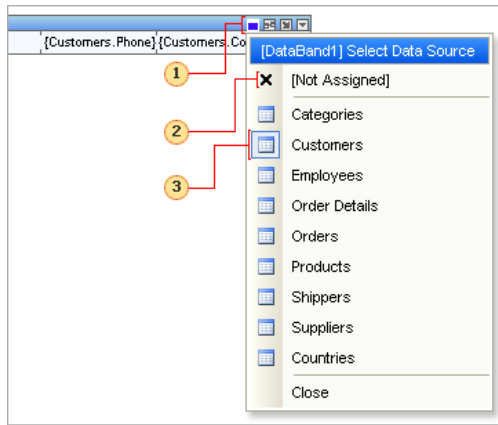
Для построения секции *Данные*, необходимо указать какой источник данных будет использоваться. Это необходимо для того, чтобы генератор отчетов знал сколько раз вывести на печать секцию *Данные*. Поэтому, в секции *Данные* делается ссылка на источник данных. Это можно сделать несколькими путями. Во-первых, можно использовать редактор секции *Данные*, для вызова которого достаточно кликнуть два раза на секцию *Данные*. Можно вызвать из контекстного меню.



Также редактор секции можно вызвать, используя свойство *Источник данных секции Данные*. Редактор секции *Данные* позволяет быстро выбрать источник данных.

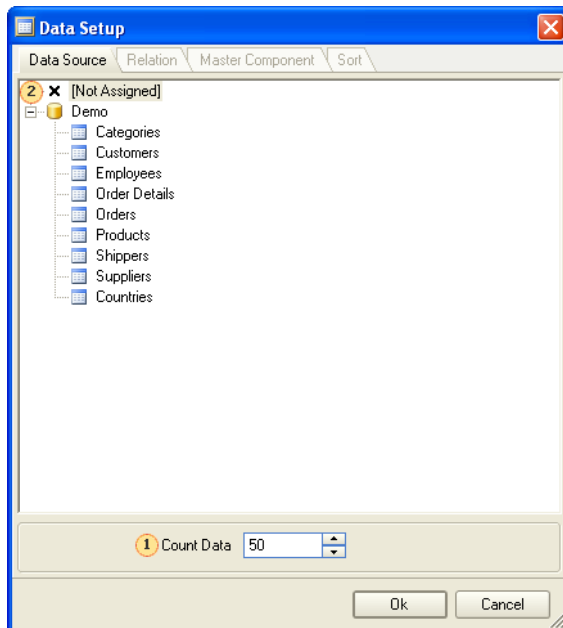


Чтобы не указывать источник данных выберите узел отмеченный на рисунке 1, Выбран источник данных *Customers* (цифра 2). Во-вторых, можно воспользоваться быстрой кнопкой на секции *Данные*, и выбрать источник данных из контекстного меню.



На рисунке выше цифрой 1 обозначена кнопка быстрого выбора источника данных. Под цифрой 2 пункт меню для не указания источника данных. Выбран источник данных *Customers* (цифра 3).

Иногда необходимо напечатать секцию *Данные* определенное количество раз, без указания ей источника данных. Для этого используется свойство *Количество данных*. По умолчанию, это свойство равно 0. Но если, например, установить его в 4, то секция *Данные* будет напечатана ровно 4 раза. Это можно использовать, например, для печати пустых граф. Стоит помнить, что в случае использования секции *Данные* как виртуальной, источник данных нужно не указывать. Указать количество элементов можно также при помощи редактора секции *Данные*.



Построим отчет, который выводит на печать список. Для этого разместим на странице одну секцию *Данные*. Секции сопоставим источник данных. На секции разместим компоненты *Текст*. В каждом компоненте *Текст* сделаем ссылку на поля данных. К примеру: `{Customers.CompanyName}`. Необходимо указать секцию *Данные*, которая будет выводить таблицу, источник данных, который используется для получения строк с данными. А также ссылку на источник данных, секции *Данные* необходимо указать источник данных, из которого необходимо брать данные и ссылки на поля источника данных. На секции *Данные* размещены компоненты *Текст*. В них сделаны ссылки на поля источника данных. При построении все ссылки будут заменены на данные. После построения отчета, все ссылки на поля данных были заменены данными из указанных полей. Секция *Данные* была выведена на печать ровно столько раз, сколько записей в источнике данных. В результате были выведены все интересующие нас поля.

Afreds Futterkiste	Obere Str. 57	030-0074321	Sales Representative
Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	(5) 555-4729	Owner
Antonio Moreno Taquería	Mataderos 2312	(5) 555-3932	Owner
Around the Horn	120 Hanover Sq.	(171) 555-7788	Sales Representative
Berglunds snabbköp	Berguvsvägen 8	0921-12 34 65	Order Administrator
Blauer See Delikatessen	Forsterstr. 57	0621-08460	Sales Representative
Blondesddd père et fils	24, place Kléber	88.60.15.31	Marketing Manager
Bólido Comidas preparadas	C/ Araquil, 67	(91) 555 22 82	Owner
Bon app'	12, rue des Bouchers	91.24.45.40	Owner
Bottom-Dollar Markets	23 Tsawassen Blvd.	(604) 555-4729	Accounting Manager
B's Beverages	Fauntleroy Circus	(171) 555-1212	Sales Representative
Cactus Comidas para llevar	Cerrito 333	(1) 135-5555	Sales Agent
Centro comercial Moctezuma	Sierras de Granada 9993	(5) 555-3392	Marketing Manager
Chop-suey Chinese	Hauptstr. 29	0462-076545	Owner
Comércio Mineiro	Av. dos Lusíadas, 23	(11) 555-7647	Sales Associate
Consolidated Holdings	Berkeley Gardens 12 Brewery	(171) 555-2282	Sales Representative
Drachenblut Delikatessen	Walsenweg 21	0241-039123	Order Administrator
Du monde entier	67, rue des Cinquante Otages	40.67.88.88	Owner

Если необходимо каждую строку данных печатать на новой странице, то можно воспользоваться свойством *Новая страница до*. Установите это свойство в *Да* и каждая строка данных будет печататься на новой странице.

Обычно в отчетах над каждым столбцом с данными, выводится наименование столбца. Для вывода наименований данных и другой информации перед данными используется специальная секция *Заголовок данных*. Она размещается на странице перед секцией *Данные*.

HeaderBand1			
Company Name	Address	Phone	Contact Title
DataBand1; Data Source: Customers			
{Customers.CompanyName}	{Customers.Address}	{Customers.Phone}	{Customers.ContactTitle}

Добавим к отчету секцию *Заголовок данных*, на которой разместим текстовые компоненты. В этих текстовых компонентах укажем наименование данных, которые выводятся на секции *Данные*. Дополнительно увеличим размер шрифта, сделаем шрифт жирным, и изменим фон у текстовых компонентов расположенных на секции *Заголовок данных*.

Company Name	Address	Phone	Contact Title
Afreds Futterkiste	Obere Str. 57	030-0074321	Sales Representative
Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	(5) 555-4729	Owner
Antonio Moreno Taquería	Mataderos 2312	(5) 555-3932	Owner
Around the Horn	120 Hanover Sq.	(171) 555-7788	Sales Representative
Berglunds snabbköp	Berguvsvägen 8	0921-12 34 65	Order Administrator
Blauer See Delikatessen	Forsterstr. 57	0621-08460	Sales Representative
Blondesddd père et fils	24, place Kléber	88.60.15.31	Marketing Manager
Bólido Comidas preparadas	C/ Araquil, 67	(91) 555 22 82	Owner
Bon app'	12, rue des Bouchers	91.24.45.40	Owner
Bottom-Dollar Markets	23 Tsawassen Blvd.	(604) 555-4729	Accounting Manager
B's Beverages	Fauntleroy Circus	(171) 555-1212	Sales Representative
Cactus Comidas para llevar	Cerrito 333	(1) 135-5555	Sales Agent
Centro comercial Moctezuma	Sierras de Granada 9993	(5) 555-3392	Marketing Manager
Chop-suey Chinese	Hauptstr. 29	0462-076545	Owner
Comércio Mineiro	Av. dos Lusíadas, 23	(11) 555-7647	Sales Associate
Consolidated Holdings	Berkeley Gardens 12 Brewery	(171) 555-2282	Sales Representative
Drachenblut Delikatessen	Walsenweg 21	0241-039123	Order Administrator
Du monde entier	67, rue des Cinquante Otages	40.67.88.88	Owner

При построении отчетов для одной секции *Данные*, можно создавать более одной секции *Заголовок данных*. К примеру, одна секция *Заголовок данных* может выводиться только в начале данных. А вторая и в начале данных и повторно на других страницах отчета. Секции *Заголовок данных* выводятся в том порядке, в каком они размещены на странице. Для одной секции *Данные*, можно создавать неограниченное количество секций *Заголовок данных*.

Если посмотреть на получившийся отчет, то можно увидеть, что генератор отчетов вывел заголовок данных повторно на всех страницах отчета. Это сделано для улучшения визуального восприятия данных. Но если это не требуется, такое поведение генератора отчетов можно изменить. Для этого можно воспользоваться свойством *Печатать на всех страницах* секции *Заголовок данных*. Установите свойство в *Нет*, и заголовок будет вводиться строго один раз перед данными. По аналогии - установите свойство *Печатать на всех страницах* в *Да*, и секция *Заголовок данных* будет выводиться на каждой странице.

При выводе списков на печать, может получиться так что заголовок будет напечатан на одной странице, а первая строка с данными на другой. Чтобы избежать такого визуального разрыва данных используйте свойство *Держать заголовок вместе* секции *Данные*. Если свойство равно *Да*, то заголовки будут печататься всегда вместе с данными. Т.е. вместе с заголовком будет выведена как минимум одна строка с данными. Если заголовок со строкой с данными не помещается на странице, то они будут перенесены на следующую страницу. Чтобы избежать вывода заголовка данных и первой строки с данными на разных страницах используйте свойство *Держать заголовок вместе*.

Часто необходимо в одном отчете *вывести несколько списков друг за другом*. Генератор отчетов никак ограничений на это не накладывает. Все что нужно сделать для построения такого отчета - это разместить на странице две секции *Данные*. Разместим на странице две секции *Данные*, укажем им разные источники данных. Кроме того, для каждой секции *Данные* создадим свой заголовок и подвал. Для этого разместим на странице по две секции *Заголовок данных* и *Итог данных*. Секции *Заголовок данных* располагаем над секцией *Данные*. Секции *Итог данных* располагаем под секцией *Данные*. При этом секции *Заголовок данных* или *Итог данных* считаются относящимися к секции *Данные* в том случае, если между этими секциями нет других секций *Данные*. К примеру, на странице расположены по две секции каждого типа. Секция *Заголовок данных1* расположен выше секции *Данные1* и между ними нет других секций *Данные*, значит, она относится к секции *Данные1*. А вот если взять секцию *Данные2*, то между ней и секцией *Заголовок данных1* находится секция *Данные1*. Поэтому, *Заголовок данных1* не относится к секции *Итог данных2*. Секция *Итог данных1* находится ниже секции *Данные1* и между двумя этими секциями нет других секций *Данные*. Значит, она относится к секции *Данные1*. А вот секция *Итог данных2* находится ниже и секции *Данные1* и секции *Данные2*. Но в случае с секцией *Данные1* между этой секцией и секцией *Итог данных2* находится секция *Данные2*. Поэтому секция *Итог данных2* относится к секции *Данные2*.

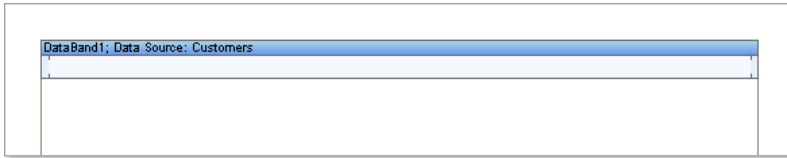
HeaderBand1			
Company	Address	Phone	Contact
DataBand1; Data Source: Customers			
{Customers.CompanyName}	{Customers.Address}	{Customers.Phone}	{Customers.ContactTitle}
FooterBand1			{Count!}
HeaderBand2			
Product	Category	Price	
DataBand2; Data Source: Products			
{Products.ProductName}	{Products.ParentCategories.CategoryName}	Products.UnitPrice	
FooterBand2			{Count!}

Первая секция *Данные* будет выводить первый список. Когда список будет полностью выведен, начнет выводиться второй список. Второй список выводит вторая секция. Количество последовательно выводимых списков в отчете не ограничено.

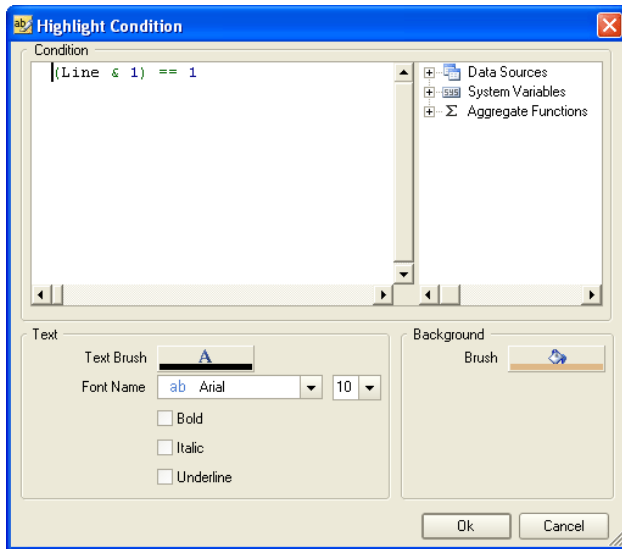
Company	Address	Phone	Contact
The Cracker Box	55 Grizzly Peak Rd.	(406) 555-5834	Marketing Assistant
Toms Spezialitäten	Luisenstr. 48	0251-031259	Marketing Manager
Tortuga Restaurante	Avda. Antea 123	(5) 555-2933	Owner
Tradição Hipermercados	Av. Inês de Castro, 414	(11) 555-2167	Sales Representative
Trail's Head Gourmet Provisioners	722 DaVinci Blvd.	(206) 555-8257	Sales Associate
Vaffeljernet	Smagsloget 45	86 21 32 43	Sales Manager
Victualles en stock	2, rue du Commerce	78.32.54.86	Sales Agent
Vins et alcools Chevalier	59 rue de l'Abbaye	26.47.15.10	Accounting Manager
Wartian Herkku	Tonikatu 38	981-443855	Accounting Manager
Wellington Importadora	Rua do Mercado, 12	(14) 555-8122	Sales Manager
White Clover Markets	305 - 14th Ave. S. Suite 3B	(206) 555-4112	Owner
Wilman Kala	Keskuskatu 45	90-224 8858	Owner/Marketing Assistant
Wolski Zajazd	ul. Filtrowa 68	(26) 642-7012	Owner
			91
Product	Category	Price	
Alice Mutton	Meat/Poultry	39	
Aniseed Syrup	Condiments	10	
Boston Crab Meat	Seafood	18,4	
Camembert Pierrot	Dairy Products	34	
Camaronvon Tigers	Seafood	62,5	
Chai	Beverages	18	

Чтобы улучшить визуальное восприятие строк с данными, можно выполнить чередующуюся заливку строк разными цветами. Это придаст отчету профессиональный вид. Для этого мож-

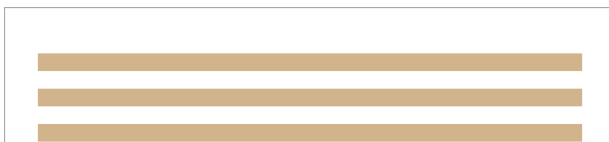
но воспользоваться условным выделением компонента *Текст*. Рассмотрим пример построения отчета с выделением каждой четной строки. Создадим новый отчет. На странице размещаем секцию *Данные*, указываем ей источник данных. Затем на секции *Данные* размещаем компонент *Текст*, так чтобы он занимал всю площадь секции.



Далее для компонента *Текст* задаем условие, согласно которому каждая четная (или нечетная) строка будет заполняться другим цветом. Для того чтобы определить является строка четной или нечетной воспользуемся системной переменной *Line*, например, так: $(Line \& 1) == 1$.



Т.е. каждую нечетную строку наше условие будет верно. Указываем цвет фона, которым будет заполняться фон компонента.

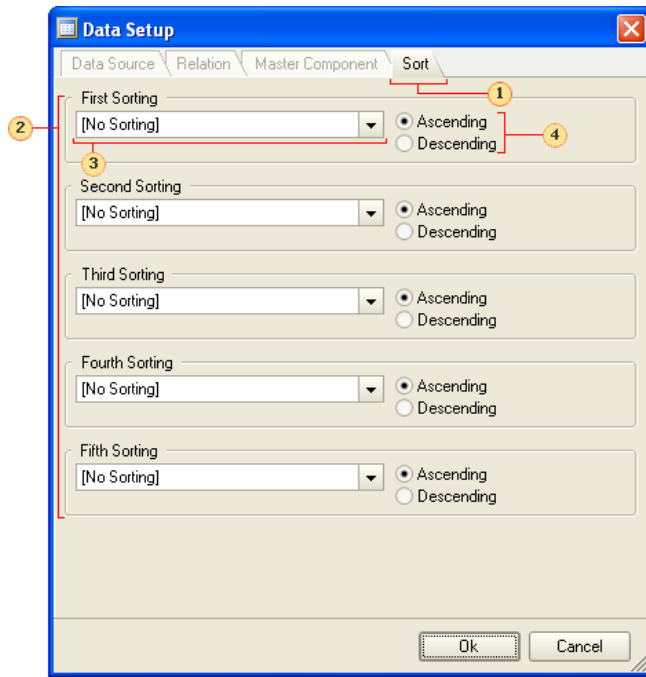


Осталось разместить компоненты с данными поверх созданного компонента *Текст*.

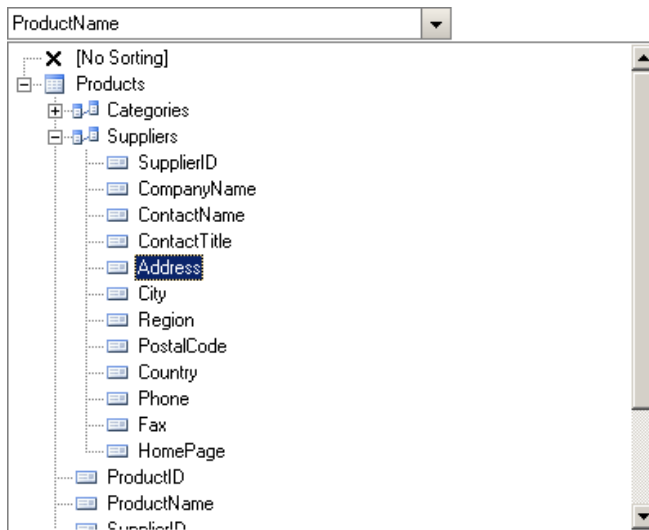
Alfreds Futterkiste	Obere Str. 57	030-0074321	Sales Representative
Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	(5) 555-4729	Owner
Antonio Moreno Taquería	Mataderos 2312	(5) 555-3932	Owner
Around the Horn	120 Hanover Sq.	(171) 555-7788	Sales Representative
Berglunds snabbköp	Berguvsvägen 8	0821-12 34 65	Order Administrator
Blauer See Delikatessen	Forsterstr. 57	0621-08460	Sales Representative

- ❖ **Важно!** Для увеличения производительности в окне предварительного просмотра, можно установить условное выделение для каждого компонента находящегося на секции.

Нередко данные, которые используются для построения отчета, отсортированы не в том порядке, который необходим. В этом случае сортировку можно выполнить средствами генератора отчетов. Сортировка задается для каждой секции *Данные* отдельно. Для настройки сортировки служит свойство *Сортировка* секции *Данные*. Допускается сортировка одновременно по пяти колонкам. Для каждой колонки можно указать порядок сортировки. Сортировать можно по возрастанию или по убыванию.



В генераторе отчетов можно отсортировать данные по колонкам из родительских связей. Для этого достаточно указать поле из родительского источника данных. Сделать это можно при настройке свойства *Сортировка*. Необходима связь между источником данных указанным для секции *Данные* и интересующим источником данных.



В генераторе отчетов можно отсортировать данные по колонкам из родительских связей.

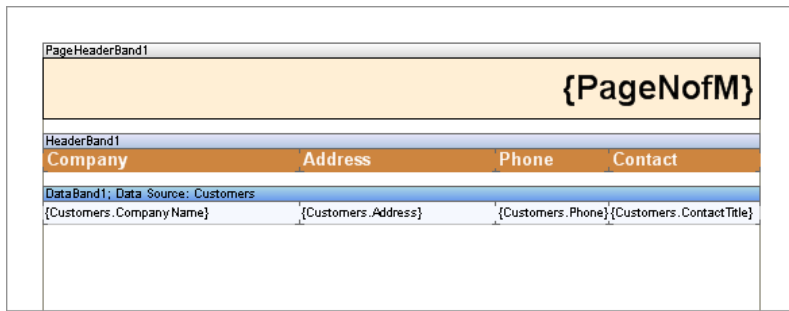
При построении отчета может возникнуть необходимость вывести не все строки из источника данных. В этом случае можно отфильтровать ненужные строки. Для этого секция *Данные* имеет два свойства *Фильтр включен* и *Фильтры*. Свойство *Фильтр включен* позволяет включать или выключать использование фильтров для секции. В свойстве *Фильтры* указывается условия печати строк. Например, условие: `{Клиенты.НазваниеФирмы.Substring(0, 1)}="А"`, позволяет напечатать список только тех клиентов, наименование которых начнется с буквы А. Кроме использования фильтра, условие можно задать при помощи событий. Для этого можно воспользоваться событием *Перед печатью*. Это событие возникает до печати компонента. Вот в этом событии и нужно написать код, который будет по определенному условию включать или выключать печать секции *Данные1*. Если условие верно, то свойство *Enabled* секции *Данные1* устанавливаем в *true*. Иначе устанавливаем свойство *Enabled* в *false*. Например: `if (Customers.CompanyName.Substring(0, 1) == "А") then Данные1.Enabled = true; else Данные1.Enabled = false;`

Секции страницы

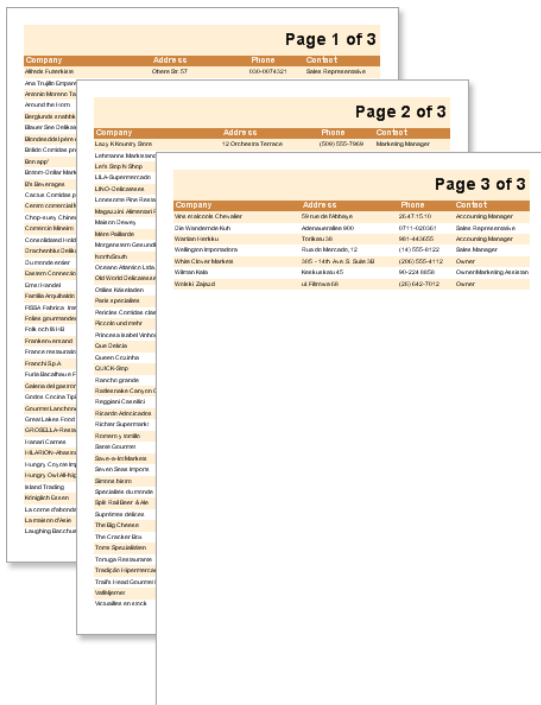
Секции страницы используются для вывода информации один раз на каждой странице. К секциям страницы относятся две секции: *Заголовок страницы* и *Итог страницы*. Секции используются для вывода заголовка страницы и подвала страницы. Секция *Заголовок страницы* является заголовком страницы. Она выводится в самом верху страницы. Секция *Итог* страницы является *подвалом* страницы. Эта секция выводится в самом низу страницы.

Заголовок страницы. Если необходимо на каждой странице отчета вывести вверху какую либо информацию, то можно воспользоваться секцией - *Заголовок Страницы*. Эта секция используется для вывода заголовка страницы. Секция *Заголовок страницы* выводится в самом верху страницы, один раз на каждой странице отчета. Секция используется для вывода номеров страниц, даты построения отчета и т.д.- любую информацию, которую нужно отображать на каждой странице. Генератор отчетов не накладывает каких либо ограничений на количество секций *Заголовок страницы*. На одной странице можно разместить два, три, десять заголовков.

Построим отчет с использованием секции *Заголовок Страницы*. На странице создадим две секции - секцию *Данные*, для вывода данных и секцию *Заголовок Данных*, для вывода заголовков данных. Далее разместим секцию *Заголовок Страницы*, на которой будем выводить номер текущей страницы и количество страниц в отчете. Для этого на секции *Заголовок страницы* создадим текстовый компонент, и в его текстовом выражении укажем выражение для вывода интересующей нас информации: *{PageNofM}*.



Запустим этот отчет на выполнение. На каждой странице вверху был выведен номер страницы.



При помощи свойства *Печатать на*, можно указать генератору отчетов, на каких страницах отчета надо выводить заголовок страницы. По умолчанию это свойство равно *Все страницы*, т.е. заголовок страницы выводится на всех страницах. Значение *Только первая страни-*

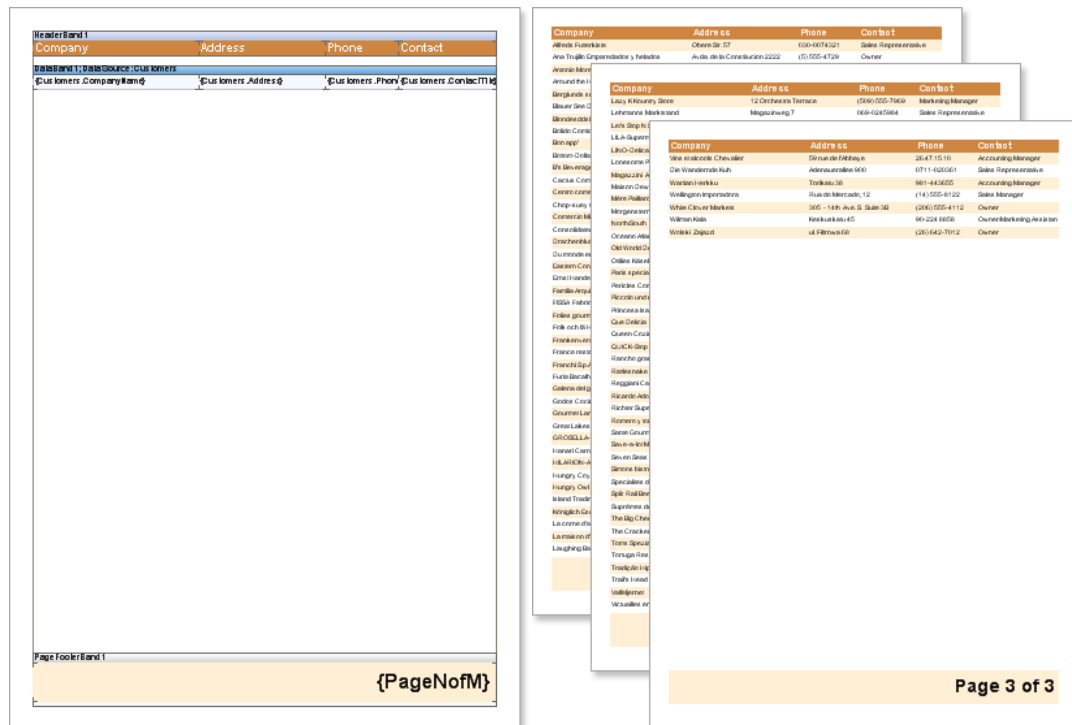
ца свойства *Печатать на* - в этом случае заголовок страницы будет выведен только на первой странице отчета.

Для вывода информации внизу каждой страницы используется секция **Итог страницы**. Секция *Итог страницы* выводится в самом низу страницы, один раз на каждой странице отчета. генератор отчетов не накладывает каких либо ограничений на количество секций *Итог страницы*. Эта секция используется для вывода номеров страниц, итогов по странице и т.д.

Построим отчет с использованием секции *Итог страницы*. На странице создадим две секции - секцию *Данные* для вывода данных, и секцию *Заголовок данных* для вывода заголовков данных. Далее разместим секцию *Итог страницы*. Стоит обратить внимание, на то что секция *Итог страницы*, в дизайнера отчетов, на странице стыкуется к нижнему краю страницы, в отличие от других секций, которые стыкуются к верхнему краю страницы.

❖ **Важно!** Секция *Итог страницы*, в дизайнера отчетов, на странице стыкуется к нижнему краю страницы.

На этой секции будем выводить номер текущей страницы и количество страниц в отчете. Создадим компонент *Текст*, выберем системную функцию *PageNoFM: {PageNoFM}*. Изображение шаблона отчета созданного с использованием секции *Итог страницы* изображено на рисунке ниже.



После построения на каждой странице внизу будет выведен номер страницы и общее количество страниц.

При помощи свойства *Печатать на*, можно указать генератору отчетов, на каких страницах выводить подвал страницы. По умолчанию это свойство равно *Все страницы*, т.е. подвал страницы выводится на каждой странице отчета. Для того чтобы запретить вывод на всех страницах кроме первой установите значение этого свойства в *Только первая страница*. В этом случае подвал страницы будет выведен только на первой странице отчета.

Одной из основных задач при построении отчетов является **задача группировки выводимых данных**. Группировка может применяться как для логического разделения строк с данными на странице, так и для визуального улучшения вида отчета. Для построения отчетов с группировкой применяется две секции – *Заголовок группы*, *Итог группы*. Секция *Заголовок группы* выводятся в начале каждой группы. Секция *Итог группы* выводится в конце каждой группы.

Simple Group

Company	Address	Phone	Contact
A			
Alfreds Futterkiste	Obere Str. 57	030-0074321	Sales Representative
Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	(5) 555-4729	Owner
Antonio Moreno Taquería	Mataderos 2312	(5) 555-3932	Owner
Around the Horn	120 Hanover Sq.	(171) 555-7788	Sales Representative
Count: 4			
B			
Berglunds snabbköp	Berguvsvägen 8	0921-12 34 65	Order Administrator
Blauer See Delikatessen	Forsterstr. 57	0621-08460	Sales Representative
Blondesddsl père et fils	24, place Kléber	88.60.15.31	Marketing Manager
Bólido Comidas preparadas	C/ Araquil, 67	(91) 555 22 82	Owner
Bon app'	12, rue des Bouchers	91.24.45.40	Owner
Bottom-Dollar Markets	23 Tsawassen Blvd.	(604) 555-4729	Accounting Manager
B's Beverages	Fauntleroy Circus	(171) 555-1212	Sales Representative
Count: 7			
C			
Cactus Comidas para llevar	Cerrito 333	(1) 135-5555	Sales Agent
Centro comercial Mochtezuma	Sierras de Granada 9993	(5) 555-3392	Marketing Manager
Chop-suey Chinese	Hauptstr. 29	0452-076546	Owner
Comércio Mineiro	Av. dos Lusíadas, 23	(11) 555-7847	Sales Associate
Consolidated Holdings	Berkeley Gardens 12 Brewery	(171) 555-2282	Sales Representative
Count: 5			

Чтобы построить отчет с группировкой, необходимо указать условие группировки. Согласно этому условию, все строки с данными будут разбиты на отдельные группы. К примеру, если выводится список компаний, то можно разбить этот список на группы, согласно первой букве в названии компании. Т.е. сначала будет идти группа *A*. В этой группе будут находиться все компании, название которых начинается с буквы *A*. Затем будет идти группа *B*. В этой группе будут выводиться все компании название которых начинается с буквы *B*. Затем будет идти группа *C*, *D*, *E* и т.д. В этом случае условием будет являться первая буква в названии компании. Следует учитывать, что генератор отчетов не сортирует автоматически строки с данными перед группировкой, поэтому сортировку следует указывать дополнительно. В нашем случае, список компаний должен быть отсортирован по колонке, которая содержит название компании. Иначе группировка будет построена неверно.

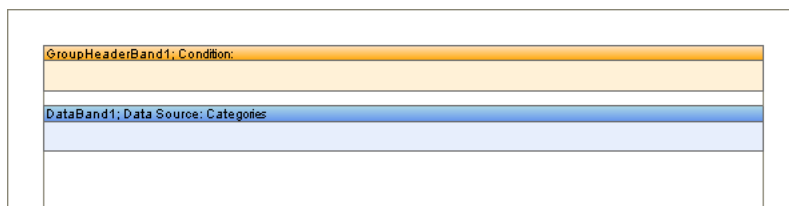
- ❖ **Важно!** Генератор отчетов не сортирует автоматически строки с данными перед группировкой, поэтому сортировку следует указывать дополнительно.

Условие задается в дизайнерах секции *Заголовок группы*. Условием может являться любое значение. К примеру, можно сгруппировать список компаний по региону в котором находятся эти компании. В этом случае *Условие* будет находиться в колонке из базы данных.

Alfreds Futterkiste		A
Ana Trujillo Emparedados y helados	→	Alfreds Futterkiste
Antonio Moreno Taquería		Ana Trujillo Emparedados y helados
Around the Horn		Antonio Moreno Taquería
Berglunds snabbköp		Around the Horn
Blauer See Delikatessen		B
Blondesddsl père et fils	→	Berglunds snabbköp
Bólido Comidas preparadas		Blauer See Delikatessen
Bon app'		Blondesddsl père et fils
Bottom-Dollar Markets		Bólido Comidas preparadas
B's Beverages	→	Bon app'
Cactus Comidas para llevar		Bottom-Dollar Markets
Centro comercial Mochtezuma		B's Beverages
Chop-suey Chinese		C
Comércio Mineiro	→	Cactus Comidas para llevar
Consolidated Holdings		Centro comercial Mochtezuma
Die Wandermde Kuh		Chop-suey Chinese
Drachenblut Delikatessen		Comércio Mineiro
Du monde entier	→	Consolidated Holdings

Заголовок группы создается при помощи секции *Заголовок группы*. Секция *Заголовок группы* является основной секцией для построения отчетов с группировкой. Без этой секции выполнить группировку невозможно. На заголовке группы размещаются компоненты для

отображения информации по группе. К примеру, это может быть наименование группы, дата, обозначение условия группировки и т.д. Секция *Заголовок группы* выводится один раз в начале каждой группы. Для создания группировки необходимо указать условие. Условие группировки задается в свойстве *Условия* этой секции или при помощи дизайнера секции *Заголовок группы*. Секция *Заголовок данных* всегда выводится выше секции *Заголовок группы*, в независимости от того, как эти секции расположены на странице в дизайнера.

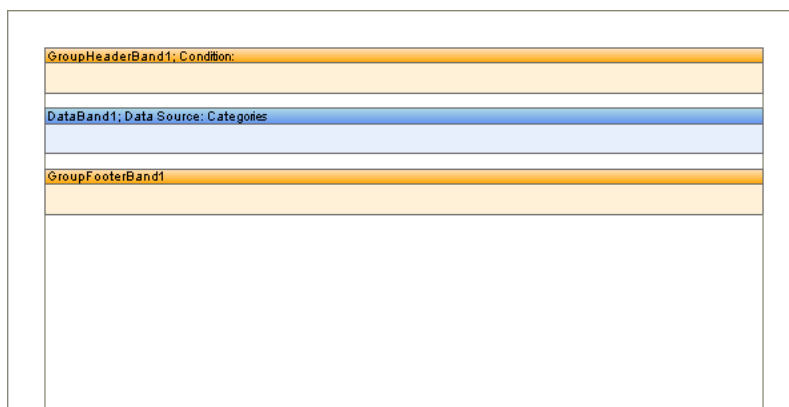


В момент построения отчета, генератор отчетов связывает заголовок группы и определенную секцию *Данные*. Секция *Заголовок группы* размещается на странице выше секции *Данные*, который выводит строки с данными. Секция *Заголовок группы* всегда относится к какой-то конкретной секции *Данные*. Обычно этой секцией является первая секция *Данные*, которая находится ниже секции *Заголовок группы*. Для построения отчета с группировкой необходима секция *Данные*.

❖ **Важно!** Для построения отчета с группировкой обязательна секция *Данные*.

Секция *Данные* необходима, потому что строки с данными выводятся при помощи этой секции и именно на основе этих строк с данными строится группировка. Также у секции *Данные* задается сортировка строк, которая влияет на порядок построения группировки.

Для вывода подвала группы используется секция *Итог группы*. Эта секция размещается ниже секции *Данные*. Причем, имеется ввиду именно та секция *Данные*, с которой связана секция *Итог группы*. Каждая секция *Итог группы*, относится к какой то определенной секции *Заголовок группы*. Без секции *Заголовок группы*, секция *Итог группы* выводится не будет.

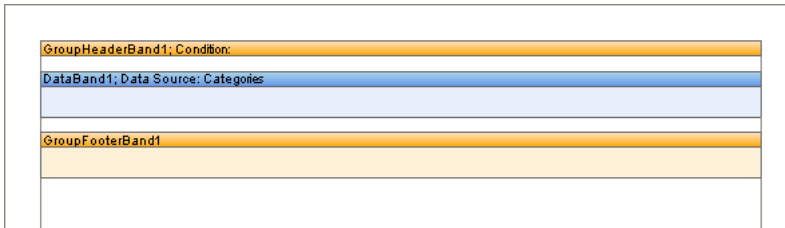


Секция *Итог группы* используется для вывода различной информации по группе. К примеру, для того чтобы вывести количество строк в группе, достаточно разместить на секции *Итог группы*, текстовый компонент. В текстовом компоненте указать выражение: `{Count()}`.

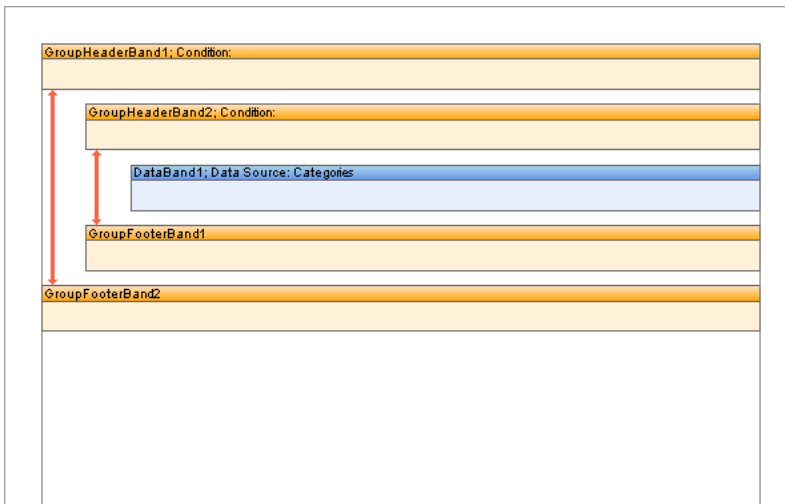
При построении отчета с группировкой, некоторые группы могут не поместиться в пределах одной страницы. Т.е. часть строк группы выведется на одной странице, а часть, в виду нехватки места, на следующей странице. Для того, чтобы избежать этого, можно заставить генератор отчетов **выводить группы вместе**. Все что необходимо сделать для этого, у секции *Заголовок группы*, к которому относится секция *Данные*, установить свойство *Держать группу вместе* в значение *Да*. В случае если у генератора не получается вывести группу целиком на одной странице, вся группа переносится на следующую страницу. Если же и на следующей странице полностью напечатать группу не удастся, то группа разрывается, и вывод производится на нескольких страницах.

Если необходимо **каждую группу выводить с новой страницы**, то у секции *Заголовок группы* достаточно установить свойство *Новая страница до* в *Да*. В случае если это свойство равно *Да*, каждая группа будет выводиться с новой страницы. Если группа не умещается на одной странице, то ее вывод будет продолжен на следующей странице. При этом следующая группа уже будет выводиться на новой странице.

Обычно, в отчетах с использованием группировки используется или только группы, или заголовки и подвалы группы одновременно. Для создания отчетов с группировкой необходимо обязательно использовать секцию *Заголовок группы*. Но в данной задаче требуется не выводить его на печать при построении отчета. В этом случае секции *Заголовок группы* необходимо установить нулевую высоту. Отчет будет успешно построен и секция *Заголовок группы* просто не будет выведена на печать, т.к. она имеет нулевую высоту.



При построении отчетов, можно использовать более одной группировки. Для этого необходимо разместить перед необходимой секцией *Данные* нужное количество секций *Заголовок группы*. После секции *Данные* размещается такое же или меньшее количество секций *Итог группы*. Каждой секции *Заголовок группы*, необходимо указать критерий группировки. При построении, секции *Заголовок группы* обрабатываются в порядке нахождения на странице, сверху вниз. Сначала самая верхняя секция, затем та, которая расположена под ней и т.д. При размещении секций *Итог группы* стоит учитывать, что последняя секция *Итог группы* соответствует первой секции *Заголовок группы*.



Рекомендуется, чтобы количество секций *Заголовок группы* и *Итог группы* было одинаковым. В случае, если секций *Итог группы* будет больше, чем секций *Заголовок группы*, будут использованы секции *Итог группы*, которые расположены ниже других, т.е. снизу вверх. Остальные секции будут проигнорированы. Если же секций *Итог группы* меньше, чем секций *Заголовок группы*, то самые нижние (которые ближе к секции *Данные*) секции *Заголовок группы* будут выведены без подвалов.

❖ **Важно!** Рекомендуется, чтобы количество секций *Заголовок группы* и *Итог группы* было одинаковым.

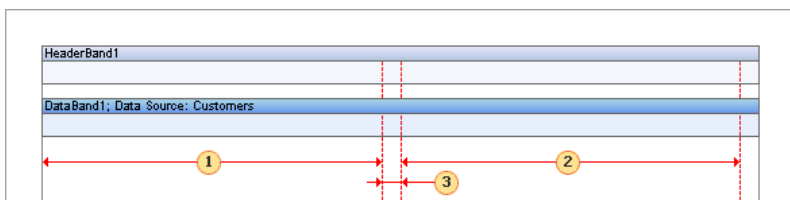
Чтобы произвести **расчет итогов по группе** достаточно разместить текстовый компонент с агрегатной функцией на секции *Заголовок группы* или *Итог группы*. К примеру, для того чтобы посчитать кол-во строк в каждой группе в текстовом компоненте Текст укажем выражение: `{Count()}`. Компонент размещается на секции *Итог группы*. После построения можно увидеть, что внизу каждой группы произведен расчет количества строк.

Для нумерации каждой строки группировки рекомендуется использовать системную переменную *Line*. Ссылка на эту системную переменную указывается в выражении текстового компонента. Текстовый компонент размещается на секции *Данные*. К примеру, разместим на секции *Данные* текстовый компонент и в его выражении укажем: `{Line}`. После построения, мы получим пронумерованный список строк в каждой группе. Нумерация начинается с 1. Когда начинается новая группа, нумерация начинается повторно с единицы. Если необходимо выполнить сквозную нумерацию, то применяется системная переменная *LineThrough*. `{LineThrough}`. В результате получается сквозная нумерация всех строк группировки.

❖ **Важно!** Во время группировки колонки на секции не могут быть использованы.

Генератор отчетов предоставляет возможность **группировать выводимые данные в колонки**. Вывод данных в виде колонок позволяет как улучшить внешний вид отчета, так и более рационально расходовать место на странице отчета. Всего поддерживается два типа вывода колонок: *колонки на странице* и *колонки на секции Данные*. Свойство Направление колонок секции *Данные* поддерживает два режима - *Вправо затем вниз* и *Вниз затем вправо*. Таким образом, генератор отчетов предоставляет полный набор инструментов для построения отчетов с колонками.

Колонки на странице. На странице отчета можно вывести данные в виде нескольких колонок. Для того чтобы включить режим вывода данных колонками, используйте свойство *Колонки страницы отчета*. По умолчанию это свойство равно 0. Если это свойство установить в значение больше чем 1, то данные будут выводиться в виде колонок. Кроме того необходимо изменить еще два свойства - *Ширина колонки* и *Расстояние между колонками*. В свойстве *Ширина колонки* указывается ширина выводимой колонки. Эта ширина действительна для всех колонок, которые будут выведены на странице. В свойстве *Расстояние между колонками* указывается расстояние между двумя колонками. Чтобы настроить страницу для вывода нескольких колонок, необходимо настроить два свойства страницы. В свойстве *Колонки* указывается количество колонок. В свойстве *Ширина колонки* указывается ширина каждой колонки.

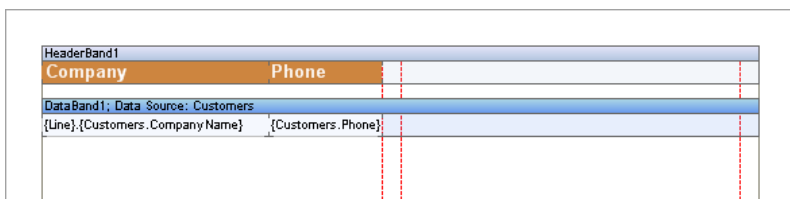


Цифры 1 и 2 - ширина первой и второй колонки соответственно. Цифра 3- расстояние между двумя колонками

В режиме вывода колонок на странице, страница как бы делится на несколько более маленьких страниц по вертикали. И отчет последовательно выводится сначала в первой части, затем во второй затем в третьей и т.д. Количество колонок на странице не ограничено.

Рассмотрим пример. Построим отчет с двумя колонками. Для этого установим свойство *Колонки* в значение 2 (т.е. две колонки). Укажем в свойстве *Ширина колонки* ширину одной колонки, а в свойстве *Расстояние между колонками* - расстояние между двумя колонками. Затем разместим на странице две секции - секцию *Данные* и секцию *Заголовок данных*. На секции *Заголовок данных* будет выводиться название данных. На секции *Данные* будут выводиться данные.

❖ **Важно!** Границы колонок указываются красной вертикальной линией.



Запустим отчет на выполнение. Как видно на рисунке ниже, на странице находится две колонки. Все линии пронумерованы.

Company	Phone	Company	Phone
1. Alfreds Futterkiste	030-0074321	46. Let's Stop N Shop	(415) 555-5938
2. Ana Trujillo Emparedados y helados	(5) 555-4729	47. LILA Supermercado	(9) 331-8954
3. Antonio Moreno Taquería	(5) 555-3932	48. LINDO-Delicatessen	(8) 34-56-12
4. Around the Horn	(171) 555-7788	49. Lonesome Pine Restaurant	(503) 555-9573
5. Berglunds snabbköp	0921-12 34 65	50. Magazzini Alimentari Riuniti	035-840230
6. Blauer See Delikatessen	0621-08460	51. Maison Dewey	(02) 201 24 67
7. Blondesddsl père et fils	88.60.15.31	52. Mère Paillarde	(514) 555-8054
8. Bóido Comidas preparadas	(91) 555 22 82	53. Morgenstem Gesundkost	0342-023176
9. Bon app'	91.24.45.40	54. North/South	(171) 555-7733
10. Bottom-Dollar Markets	(804) 555-4729	55. Océano Atlántico Ltda.	(1) 135-5333
11. B's Beverages	(171) 555-1212	56. Old World Delicatessen	(907) 555-7584
12. Cactus Comidas para llevar	(1) 135-5555	57. Otilies Käseladen	0221-0644327
13. Centro comercial Motezuma	(5) 555-3392	58. Paris spécialités	(1) 42.34.22.66
14. Chop-suey Chinese	0452-076545	59. Pericles Comidas clásicas	(5) 552-3745
15. Comércio Mineiro	(11) 555-7647	60. Piccolo umh mehr	6562-9722
16. Consolidated Holdings	(171) 555-2282	61. Princesa Isabel Vinhos	(1) 356-5634
17. Die Wandernde Kuh	0711-020361	62. Que Delicia	(21) 555-4252
18. Drachenblut Delikatessen	0241-039123	63. Queen Cozinha	(11) 555-1189
19. Du monde entier	40.67.88.88	64. QUICK-Stop	0372-035188
20. Eastem Connection	(171) 555-0287	65. Rancho grande	(11) 123-5555

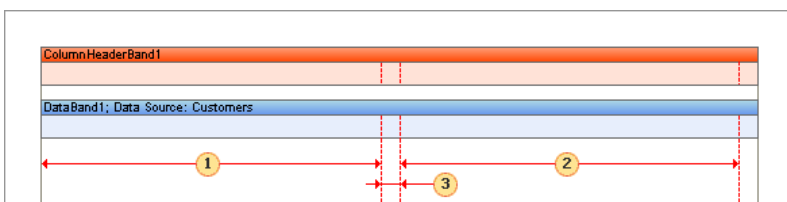
Вывод колонок отчета происходит следующим образом. Генератор отчетов печатает секции до тех пор, пока не закончится место на странице. После этого, вместо формирования новой страницы, он добавляет с правой стороны новую колонку. После чего продолжает печать данных сверху. Так происходит пока не будут напечатаны все данные или пока не кончится все место на странице.

Multi Column List

Company	Phone	Company	Phone
1. Alfreds Futterkiste	030-0074321	42. La maison d'Asie	61 37 61 10
2. Ana Trujillo Emparedados y helados	(5) 555-4729	43. Laughing Buddha Wine Cellars	(604) 555-5592
3. Antonio Moreno Taquería	(5) 555-3932	44. Lucy's Honey Buns	(206) 555-7659
4. Around the Horn	(171) 555-7788	45. Lufthansa Marktstand	069-024564
5. Berglunds snabbköp	0921-12 34 65	46. Let's Stop N Shop	(415) 555-5938
6. Blauer See Delikatessen	0621-08460	47. LILA Supermercado	(9) 331-8954
7. Blondesddsl père et fils	88.60.15.31	48. LINDO-Delicatessen	(8) 34-56-12
8. Bóido Comidas preparadas	(91) 555 22 82	49. Lonesome Pine Restaurant	(503) 555-9573
9. Bon app'	91.24.45.40	50. Magazzini Alimentari Riuniti	035-840230
10. Bottom-Dollar Markets	(804) 555-4729	51. Maison Dewey	(02) 201 24 67
11. B's Beverages	(171) 555-1212	52. Mère Paillarde	(514) 555-8054
12. Cactus Comidas para llevar	(1) 135-5555	53. Morgenstem Gesundkost	0342-023176
13. Centro comercial Motezuma	(5) 555-3392	54. North/South	(171) 555-7733
14. Chop-suey Chinese	0452-076545	55. Océano Atlántico Ltda.	(1) 135-5333
15. Comércio Mineiro	(11) 555-7647	56. Old World Delicatessen	(907) 555-7584
16. Consolidated Holdings	(171) 555-2282	57. Otilies Käseladen	0221-0644327
17. Die Wandernde Kuh	0711-020361	58. Paris spécialités	(1) 42.34.22.66
18. Drachenblut Delikatessen	0241-039123	59. Pericles Comidas clásicas	(5) 552-3745
19. Du monde entier	40.67.88.88	60. Piccolo umh mehr	6562-9722
20. Eastem Connection	(171) 555-0287	61. Princesa Isabel Vinhos	(1) 356-5634
21. El Manicomio	3835-5425	62. Que Delicia	(21) 555-4252
22. Farma del Acqua	(11) 555-4673	63. Queen Cozinha	(11) 555-1189
23. FICHA FATECA	(61) 555-94	64. QUICK-Stop	0372-035188
24. FINE gourmet	20 16 10	65. Rancho grande	(11) 123-5555
25. FINEST BIRRA	6699 77 21	66. Restaurant Campana Gastronomia	(55) 555-5659
26. France restauration	41 21 21	67. Riggieri Cacialci	052 250791
27. French SpA	1-888-0260	68. Ricardo's Abocadito	(21) 555-3412
28. Fucina veneta	169-647310	69. Richter Supermarkt	066 244214
29. G. Galambos	(1) 555-2524	70. Ricos y rorle	491 50260
30. Galeries Lafayette	1601 201 4560	71. Sana's Gourmet	071 32 35
31. Gander Cookies	1601 555 82 82	72. Saw-ee-Blackies	(20) 555-8667
32. Gammarelly	(11) 555-4682	73. Sela's Super Impres	(171) 555-1117
33. Grand Larousse	(604) 555-7000	74. Serrano's	34 34 56
34. GROSJOLLA S.p.A.	02 281-2051	75. Spicceria diumonde	(41) 555-6110
35. Gurney's Catering	(2) 555-4961	76. Sph. Holdbar & Co	(31) 555-6889
36. GUYTON-BOURNE	(5) 555-1349	77. Supercasa delicias	(61) 715 07 22 20
37. Hungry Cityzen Importers	(555) 555-6874	78. The Big Cheese	(504) 555-3412
38. Hungry Owl All Right Concern	2967 542	79. The Cracker Box	(440) 555-5834
39. Idealized Trading	(180) 555-8888	80. Tora's Specialties	0201-610259
40. Innoventi Group	055 555-6876	81. Torque Restaurant	(5) 555-2963
41. La Casa del Abundancia	361584-10	82. Tradici Supermercado	(11) 555-2147

(c) 2003-2006 eSimul soft Page 1 of 2

Колонки на секции Данные. Вывод колонок на странице имеет один недостаток. **Данных** может хватить только на одну колонку. В результате, остальные колонки могут быть незаполненными. Это способствует нерациональному использованию места на странице. Для того чтобы обойти эту проблему, можно воспользоваться возможностью вывода колонок при помощи секции **Данные**. Для того чтобы включить режим вывода данных колонками, используется свойство **Колонки** секции **Данные**. Установите этому свойству значение более 1. Также как и в случае с колонками на странице требуется установить два свойства - **Ширина колонки** и **Расстояние между колонками**. В свойстве **Ширина колонки** указывается ширина выводимой колонки. Эта ширина действительна для всех колонок, которые будут выведены на секции **Данные**. В свойстве **Расстояние между колонками** указывается расстояние между двумя колонками.



❖ **Важно!** Количество колонок на секции **Данные** не ограничено.

Колонки на секции *Данные* могут быть выведены в двух режимах: *Вправо затем вниз* и *Вниз затем вправо*. Режим выбирается в свойстве *Направление колонок*.

Режим вывода колонок – Вправо затем вниз. В этом режиме секции *Данные*, значения выводятся последовательно по колонкам, слева направо. Когда все значения будут выведены в колонках, формируется новая строка и на ней опять выводятся все значения по колонкам. Таким образом, данные, выводимые при помощи колонок, займут на листе бумаги ровно столько места сколько им необходимо.

Company	Company	Company
1. Alfredo Futterkiste	2. Ana Trujillo Emparedados y helados	3. Antonio Moreno Taquería
4. Around the Horn	5. Berglunds snabbköp	6. Blauer See Delikatessen
7. Blondesddsl père et fils	8. Bóldo Comidas preparadas	9. Bon app'
10. Bottom-Dollar Markets	11. B's Beverages	12. Cactus Comidas para llevar
13. Centro comercial Moctezuma	14. Chop-suey Chinese	15. Comércio Mineiro
16. Consolidated Holdings	17. Die Wandermde Kuh	18. Drachenblut Delikatessen
19. Du monde entier	20. Eastem Connection	21. Ernst Handel
22. Familia Arquibaldo	23. FISSA Fabrica Inter. Salchichas S.A	24. Folies gourmandes
25. Folk och få HB	26. Franoe restauration	27. Franchi S.p.A.
28. Frankenversand	29. Furtu Bacalhau e Frutos do Mar	30. Galeria del gastrónomo
31. Godos Cocina Típica	32. Gourmet Lanchonetes	33. Great Lakes Food Market
34. GROSELLA-Restaurante	35. Hanari Carnes	36. HILARION-Abastos
37. Hungry Coyote Import Store	38. Hungry Owl All-Night Grocers	39. Island Trading
40. Königlich Essen	41. La come d'abondance	42. La maison d'Asie
43. Laughing Bacchus Wine Cellars	44. Lazy K Kountry Store	45. Lehmanns Marktstand
46. Let's Stop N Shop	47. LILA-Supermercado	48. LINO-Delicateses
49. Lonesome Pine Restaurant	50. Magazzini Alimentari Riuniti	51. Maison Dewey
52. Mère Pailarde	53. Morgenstern Gesundkost	54. North/South
55. Océano Atlántico Ltda.	56. Old World Delicatessen	57. Otilies Käseladen
58. Paris spécialités	59. Peroles Comidas clásicas	60. Piccolo und mehr
61. Pinesza Isabel Vnhos	62. Que Delícia	63. Queen Cozinha
64. Pilsener Beer	64. Pilsener Beer	64. Pilsener Beer
65. Pilsener Beer	65. Pilsener Beer	65. Pilsener Beer
66. Pilsener Beer	66. Pilsener Beer	66. Pilsener Beer
67. Pilsener Beer	67. Pilsener Beer	67. Pilsener Beer
68. Pilsener Beer	68. Pilsener Beer	68. Pilsener Beer
69. Pilsener Beer	69. Pilsener Beer	69. Pilsener Beer
70. Pilsener Beer	70. Pilsener Beer	70. Pilsener Beer
71. Pilsener Beer	71. Pilsener Beer	71. Pilsener Beer
72. Pilsener Beer	72. Pilsener Beer	72. Pilsener Beer
73. Pilsener Beer	73. Pilsener Beer	73. Pilsener Beer
74. Pilsener Beer	74. Pilsener Beer	74. Pilsener Beer
75. Pilsener Beer	75. Pilsener Beer	75. Pilsener Beer
76. Pilsener Beer	76. Pilsener Beer	76. Pilsener Beer
77. Pilsener Beer	77. Pilsener Beer	77. Pilsener Beer
78. Pilsener Beer	78. Pilsener Beer	78. Pilsener Beer
79. Pilsener Beer	79. Pilsener Beer	79. Pilsener Beer
80. Pilsener Beer	80. Pilsener Beer	80. Pilsener Beer
81. Pilsener Beer	81. Pilsener Beer	81. Pilsener Beer
82. Pilsener Beer	82. Pilsener Beer	82. Pilsener Beer
83. Pilsener Beer	83. Pilsener Beer	83. Pilsener Beer
84. Pilsener Beer	84. Pilsener Beer	84. Pilsener Beer
85. Pilsener Beer	85. Pilsener Beer	85. Pilsener Beer
86. Pilsener Beer	86. Pilsener Beer	86. Pilsener Beer
87. Pilsener Beer	87. Pilsener Beer	87. Pilsener Beer
88. Pilsener Beer	88. Pilsener Beer	88. Pilsener Beer
89. Pilsener Beer	89. Pilsener Beer	89. Pilsener Beer
90. Pilsener Beer	90. Pilsener Beer	90. Pilsener Beer
91. Pilsener Beer	91. Pilsener Beer	91. Pilsener Beer

Количество выводимых колонок на секции *Данные* не ограничено. Построим отчет с колонками. Построим отчет с тремя колонками на секции *Данные*. Для этого на странице создадим две секции - секцию *Данные* и секцию *Заголовок колонки*. После чего установим свойство *Колонки* в значение 3 (т.е. три колонки). Укажем в свойстве *Ширина колонки* ширину одной колонки, а в свойстве *Расстояние между колонками* - расстояние между колонками. На секции *Заголовок колонки* расположим текстовые компоненты, в них будут выводиться наименования колонок. На секции *Данные* будут выводиться данные. Границы колонок указываются красной вертикальной линией. Кроме того, все компоненты, расположенные на первой колонке, в момент редактирования проецируются на другие колонки.

ColumnHeaderBand1
Company
DataBand1; Data Source: Customers
{Line}. {Customers.CompanyName}

Запускаем отчет на выполнение. По нумерации строк может легко увидеть направление вывода колонок в отчете.

Company	Company	Company
1. Alfredo Futterkiste	2. Ana Trujillo Emparedados y helados	3. Antonio Moreno Taquería
4. Around the Horn	5. Berglunds snabbköp	6. Blauer See Delikatessen
7. Blondesddsl père et fils	8. Bóldo Comidas preparadas	9. Bon app'
10. Bottom-Dollar Markets	11. B's Beverages	12. Cactus Comidas para llevar
13. Centro comercial Moctezuma	14. Chop-suey Chinese	15. Comércio Mineiro
16. Consolidated Holdings	17. Die Wandermde Kuh	18. Drachenblut Delikatessen
19. Du monde entier	20. Eastem Connection	21. Ernst Handel
22. Familia Arquibaldo	23. FISSA Fabrica Inter. Salchichas S.A	24. Folies gourmandes
25. Folk och få HB	26. Franoe restauration	27. Franchi S.p.A.
28. Frankenversand	29. Furtu Bacalhau e Frutos do Mar	30. Galeria del gastrónomo
31. Godos Cocina Típica	32. Gourmet Lanchonetes	33. Great Lakes Food Market
34. GROSELLA-Restaurante	35. Hanari Carnes	36. HILARION-Abastos
37. Hungry Coyote Import Store	38. Hungry Owl All-Night Grocers	39. Island Trading
40. Königlich Essen	41. La come d'abondance	42. La maison d'Asie
43. Laughing Bacchus Wine Cellars	44. Lazy K Kountry Store	45. Lehmanns Marktstand
46. Let's Stop N Shop	47. LILA-Supermercado	48. LINO-Delicateses
49. Lonesome Pine Restaurant	50. Magazzini Alimentari Riuniti	51. Maison Dewey
52. Mère Pailarde	53. Morgenstern Gesundkost	54. North/South
55. Océano Atlántico Ltda.	56. Old World Delicatessen	57. Otilies Käseladen
58. Paris spécialités	59. Peroles Comidas clásicas	60. Piccolo und mehr
61. Pinesza Isabel Vnhos	62. Que Delícia	63. Queen Cozinha

Режим вывода колонок – Вниз затем вправо. Вывод колонок в режиме *Вправо затем вниз* имеет существенный недостаток. Информацию достаточно неудобно читать со страницы, т.к. данные выводятся слева направо, а потом вниз. Куда более привычным является режим вывода колонок, который обеспечивает *Вниз затем вправо*. Этот режим существенно отличается от режима *Вправо затем вниз*. Сначала целиком выводится первая колонка, затем целиком вторая и т.д.

Company	Company	Company
1. ABC Inc. Fullerton	32. Gourmet Lunches	63. Queen Cozinha
2. Ana Trujillo Emparedados y Helado	33. Great Lakes Food Market	64. O.K. Kebab
3. Antonio Moreno Taqueria	34. GRAND BELLARYS	65. Rancho grande
4. Arora Inc. (New York)	35. Harbin Coffee	66. Rattlesnake (Beverly Hills)
5. Berglunds snabbkop	36. HILARIO M. Aulas	67. Reggiani Cafe
6. Blauer Dre Belles	37. Hungry Howie's	68. Riccioli's
7. Bonifazi's deli	38. Hungry Howie's	69. Richter's
8. Bonolis Concessionaries	39. Italian Delicacies	70. Roma's
9. Bonapp'	40. Italian Delicacies	71. Sbarro
10. Bolton-Butler	41. L.A. Lime Delicacies	72. Sbarro
11. B's Burgers	42. L.A. Lime Delicacies	73. Seven Dials
12. Baciotti Confectionery	43. L.A. Lime Delicacies	74. Simons
13. Centrico Confectionery	44. L.A. Lime Delicacies	75. Spiceland's
14. Chop-suey Chinese	45. L.A. Lime Delicacies	76. Spiceland's
15. Confiserie Moretti	46. L.A. Lime Delicacies	77. Spiceland's
16. Concessionaries	47. L.A. Lime Delicacies	78. The Big One
17. D&M Manufacturing	48. L.A. Lime Delicacies	79. The Crackers
18. Drachmann's Deli	49. L.A. Lime Delicacies	80. Tom's Specialties
19. Duromatic Enterprises	50. L.A. Lime Delicacies	81. Tortuga Restaurant
20. Eastman Confectionery	51. L.A. Lime Delicacies	82. Truffles
21. Eats 'n' Treats	52. L.A. Lime Delicacies	83. Tully's
22. Familia Arcautolo	53. L.A. Lime Delicacies	84. Vanille
23. F&B Fabrications	54. L.A. Lime Delicacies	85. Vivaldi's
24. F&B's Confectionery	55. L.A. Lime Delicacies	86. Vivaldi's
25. F&B's Confectionery	56. L.A. Lime Delicacies	87. Vivaldi's
26. France's Burgers	57. L.A. Lime Delicacies	88. Vivaldi's
27. French Sp. A.	58. L.A. Lime Delicacies	89. Vivaldi's
28. French Confectionery	59. L.A. Lime Delicacies	90. Vivaldi's
29. F&B's Confectionery	60. L.A. Lime Delicacies	91. Vivaldi's
30. G&H's Confectionery	61. L.A. Lime Delicacies	92. Vivaldi's
31. G&H's Confectionery	62. L.A. Lime Delicacies	

В режиме *Вниз затем вправо* генератор отчетов старается поровну распределить все значения между доступными колонками. После того как он распределил строки между колонками, начинается вывод первой колонки. Причем колонка выводится не до конца страницы, а так, чтобы вместить то количество элементов, которые предназначены для этой колонки. После вывода первой начинается вывод второй колонки и т.д. Таким образом, данные, выводимые при помощи колонок, займут на листе бумаги ровно столько места сколько им необходимо. Кроме того, данные будут распределены приблизительно поровну между всеми колонками (в отличие от вывода колонок на странице). И все данные будут представлены на листе в удобной форме (в отличие от режима *Вправо затем вниз*). Количество выводимых колонок на секции *Данные* неограничено.

Построим отчет с выводом колонок в режиме *Вниз затем вправо*. Создаем на странице две секции - секцию *Данные* и секцию *Заголовок колонки*. После чего устанавливаем свойство *Колонки* в значение 3 (т.е. три колонки). В свойстве *Ширина колонки* указываем ширину одной колонки, а в свойстве *Расстояние между колонками* - расстояние между колонками. Границы колонок указываются красной вертикальной линией. Кроме того, все компоненты, расположенные на первой колонке, в момент редактирования проецируются на другие колонки.

Company		
{Line}. {Customers.Company Name}		

Запускаем отчет на выполнение. Генератор отчетов постарался поровну распределить строки между всеми тремя колонками. В первой колонке поместилось 31 строка, во второй также 31, а в третьей 29. Вся информация подана на странице в удобном виде, сверху вниз, слева направо.

Company	Company	Company
1. Alfreds Futterkiste	32. Gourmet Lanchonetes	63. Queen Cozinha
2. Ana Trujillo Emparedados y helados	33. Great Lakes Food Market	64. QUICK-Stop
3. Antonio Moreno Taquería	34. GROSELLA-Restaurante	65. Rancho grande
4. Around the Horn	35. Hanari Carnes	66. Rattlesnake Canyon Grocery
5. Berglunds snabbköp	36. HILARION-Abastos	67. Reggiani Caseifici
6. Blauer See Delikatessen	37. Hungry Coyote Import Store	68. Ricardo Adocicados
7. Blondesdél père et fils	38. Hungry Owl All-Night Grocers	69. Richter Supermarkt
8. Bólido Comidas preparadas	39. Island Trading	70. Romero y tomillo
9. Bon app'	40. Königlich Essen	71. Santé Gourmet
10. Bottom-Dollar Markets	41. La cornue d'abondance	72. Save-a-lot Markets
11. B's Beverages	42. La maison d'Asie	73. Seven Seas Imports
12. Cactus Comidas para llevar	43. Laughing Bacchus Wine Cellars	74. Simons bistro
13. Centro comercial Motezuma	44. Lazy K Kountry Store	75. Spécialités du monde
14. Chop-suey Chinese	45. Lehmanns Marktstand	76. Split Rail Beer & Ale
15. Comércio Mineiro	46. Let's Stop N Shop	77. Suprêmes délices
16. Consolidated Holdings	47. LILA-Supermercado	78. The Big Cheese
17. Die Wandermühle	48. LINO-Delicatessen	79. The Cracker Box
18. Drachenblut Delikatessen	49. Lonesome Pine Restaurant	80. Toms Spezialitäten
19. Du monde entier	50. Magazzini Alimentari Riuniti	81. Tortuga Restaurante
20. Eastern Connection	51. Maison Dewey	82. Tradição Hipermercados
21. Ernst Handel	52. Mère Poularde	83. Trail's Head Gourmet Provisioners
22. Família Arqubaldo	53. Morgenstern Gesundkost	84. Varfjellet
23. FISSA Fabrica Inter. Salchichas S.A	54. North/South	85. Vœuilles en stock
24. Folies gourmandes	55. Océano Atlântico Ltda.	86. Vns et alcools Chevalier
25. Folk och få HB	56. Old World Delicatessen	87. Wartian Herkku
26. France restauration	57. Ottilies Käseladen	88. Wellington Importadora
27. Franchi S.p.A.	58. Paris spécialités	89. White Clover Markets
28. Frankenversand	59. Peñoles Comidas clásicas	90. Wilman Kala
29. Furia Bacalhau e Frutos do Mar	60. Piccolo und mehr	91. Wolski Zajazd
30. Galería del gastrónomo	61. Princesa Isabel Vinhos	
31. Godos Cocina Típica	62. Que Delícia	

При использовании режима вывода колонок *Вниз затем вправо*, может возникнуть следующая ситуация. В отчете выводится слишком мало строк, к примеру пять. При этом данные будут распределены поровну между всеми колонками. В некоторых случаях необходимо, чтобы генератор отчетов не распределял поровну между всеми колонками строки с данными. А для лучшей визуальной подачи данных, лучше было бы если бы все строки с данными были бы выведены в одной колонке.

Name	Name
1. Alice Mutton	4. Perth Pasties
2. Mshi Kobe Niku	5. Thüringer Rostbratwurst
3. Pâté chinois	6. Tourtière
1. Longlife Tofu	4. Tofu
2. Manjimup Dried Apples	5. Uncle Bob's Organic Dried Pears
3. Rössle Sauerkraut	

Вот для этого случая и предназначено специальное свойство секции *Данные - Минимальное количество строк в колонке*. В этом свойстве указывается минимальное количество строк в первой колонке. По умолчанию свойство равно 0, что значит что проверка на минимальное количество строк не делается. Если значение этого свойства больше нуля, то в первой колонке будет выведено количество строк не менее указанного в этом свойстве.

Name	Name
1. Alice Mutton	6. Tourtière
2. Mshi Kobe Niku	
3. Pâté chinois	
4. Perth Pasties	
5. Thüringer Rostbratwurst	
1. Longlife Tofu	
2. Manjimup Dried Apples	
3. Rössle Sauerkraut	
4. Tofu	
5. Uncle Bob's Organic Dried Pears	

Чтобы вывести заголовки данных, обычно используется секция *Заголовок данных*. Однако, для колонок существует специальная секция - *Заголовок колонки*. Отличие состоит в следующем - секция *Заголовок данных* выводится один раз перед секцией *Данные*. Секция *Заголовок колонки* также выводится один раз, но компоненты, расположенные на ней, будут повторно выведены над каждой колонкой. Причем секция *Заголовок колонки* используется

много по другим правилам. Сначала выводятся все секции *Итог колонки*. Затем все секции *Итог данных* в случае, если выведены все строки с данными. Но если у секции *Итог данных* свойство *Печатать на всех страницах* установлено в *Да*, то секции выводятся в порядке расположения на странице. Следует также учитывать, что если свойство *Печатать на всех страницах* секции *Итог данных* не установлено в *Да*, то эта секция будет выведена только после вывода всех строк. Это необходимо учитывать при построении отчетов.

Header1		
Column Header1	Column Header1	Column Header1
Header2		
Column Header2	Column Header2	Column Header2
1.Data	2.Data	3.Data
4.Data	5.Data	6.Data
7.Data	8.Data	9.Data
10.Data	11.Data	12.Data
13.Data	14.Data	15.Data
16.Data	17.Data	18.Data
19.Data		
Column Footer1	Column Footer1	Column Footer1
Column Footer2	Column Footer2	Column Footer2
Footer1		

Режим вывода колонок - Вниз затем вправо. В этом режиме вывод секций заголовков полностью аналогичен выводу в режиме *Вправо затем вниз*. Т.е. все секции выводятся в том порядке, в котором они расположены на странице. В случае с подвалами, если у секции *Итог данных* установлено свойство *Печатать на всех страницах* в *Да*, то все секции подвалов выводятся в порядке расположения на странице. Если не установлено, то на печать выводятся только секции *Итог колонки*. Секции *Итог данных* в этом случае игнорируются.

Header1		
Column Header1	Column Header1	Column Header1
Header2		
Column Header2	Column Header2	Column Header2
1.Data	8.Data	15.Data
2.Data	9.Data	16.Data
3.Data	10.Data	17.Data
4.Data	11.Data	18.Data
5.Data	12.Data	19.Data
6.Data	13.Data	
7.Data	14.Data	
Column Footer1	Column Footer1	Column Footer1
Column Footer2	Column Footer2	Column Footer2

Вывод колонок на странице. При выводе колонок на странице, секция *Заголовок колонки* эквивалентна секции *Заголовок данных*, а секция *Итог колонки* эквивалентна секции *Итог данных*. Единственная разница, в том что секции *Заголовок колонки* и *Итог колонки* выводятся для каждой колонки. Все секции выводятся в порядке расположения на странице.

Свойство Тип Автоширины (AutoWidthType) таблицы определяет как генератор отчетов будет править размеры ячеек после построения отчета. Если выбрано **Нем (None)**, то ширина колонок подбирается в зависимости от содержимого ячеек всей таблицы (берется самая длинная строка по колонке). Если у колонки установлено свойство *Фиксированная ширина (FixedWidth)*, то ее ширина не изменяется.

Bodos Coolins Tipicos	C/ Romero, 33	(86) 666 82 82	Sales Manager
Gourmet Lanchonetes	Av. Brasil, 442	(11) 666-8482	Sales Associate
Great Lakes Food Market	2712 Baker Blvd.	(603) 666-7666	Marketing Manager
GROBELLA-Restaurante	8ª Ave. Loc Palos Grandes	(2) 283-2861	Owner

Company	Address	Phone	ContactTitle
Hanari Carnes	Rua do Paço, 87	(21) 666-0081	Accounting Manager
HILARION-Abastos	Carrera 22 con Ave. Carlos 8:(6) 666-1040		Sales Representative
Hungry Coyote Import Store	City Center Plaza 618 Main St.	(603) 666-8874	Sales Representative
Hungry Owl All-Night Grocers	8 Johnstown Road	2987 542	Sales Associate
Island Trading	Green House, Crowther Way	(185) 666-8888	Marketing Manager

Если выбрано - **Полная таблица (FullTable)**, то ширина колонок подбирается в зависимости от ширины таблицы, т.е. сначала проверяется ширина всех ячеек колонки целой таблицы (колонке устанавливается максимальное значение), после чего, если остается свободное место, то оно распределяется между колонками, у которых свойство *FixedWidth=false*, если места не хватает, то от всех колонок, у которых свойство *FixedWidth=false*, отнимается равное количество ширины колонки, чтобы все колонки поместились.

Dodos Coolina Tipicos	C/ Romero, 33	(86) 666 32 32	Sales Manag
Gourmet Lanchonetes	Av. Brasil, 442	(11) 666-8432	Sales Assoc
Great Lakes Food Market	2732 Baker Blvd.	(603) 666-7666	Marketing M.
GROSELLA-Restaurant	5ª Ave. Los Palos Grandes	(2) 283-2861	Owner

Company	Address	Phone	ContactT
Hanari Carnes	Rua do Paço, 87	(21) 666-0081	Accounting M
HILARION-Abastos	Carrera 22 con Ave. Carlos Soubelle #8-35	(6) 666-1040	Sales Repre
Hungry Coyote Import Store	City Center Plaza 618 Main St.	(603) 666-8874	Sales Repre
Hungry Owl All-Night Grocers	8 Johnstown Road	2987 542	Sales Assoc

Если выбрано - **Последние колонки (LastColumns)**, то ширина колонок подбирается в зависимости от ширины таблицы, т.е. сначала проверяется ширина всех ячеек колонки целой таблицы (колонке устанавливается максимальное значение), после чего, если остается свободное место, то оно прибавляется к последней колонке, у которой свойство *FixedWidth=false*. Если места не хватает, то недостающее место отнимается от последних колонок, у которых свойство *FixedWidth=false*.

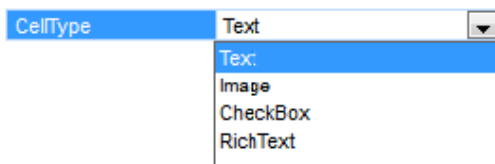
Dodos Coolina Tipicos	C/ Romero, 33	(86) 666 32 32	Sales Manager
Gourmet Lanchonetes	Av. Brasil, 442	(11) 666-8432	Sales Associate
Great Lakes Food Market	2732 Baker Blvd.	(603) 666-7666	Marketing Manager
GROSELLA-Restaurant	5ª Ave. Los Palos Grandes	(2) 283-2861	Owner

Company	Address	Phone	ContactTitle
Hanari Carnes	Rua do Paço, 87	(21) 666-0081	Accounting Manager
HILARION-Abastos	Carrera 22 con Ave. Carlos Soubelle #8-35	(6) 666-1040	Sales Representative
Hungry Coyote Import Store	City Center Plaza 618 Main St.	(603) 666-8874	Sales Representative
Hungry Owl All-Night Grocers	8 Johnstown Road	2987 542	Sales Associate

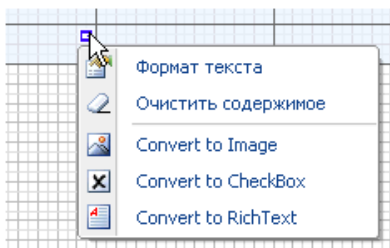
Свойство **Фиксированная ширина**. В компоненте таблица могут находиться ячейки различных видов: текст, изображение, флажок и *Rtf* текст. Каждый из видов ячейки отображается по-разному и соответствует определенному компоненту генератора отчетов:

- **Текст** - ячейка будет отображаться как текст. Настройки ячейки аналогичны настройкам компонента *Текст (Text)*;
- **Изображение** - ячейка будет отображаться как картинка. Настройки ячейки аналогичны настройкам компонента *Картинка (Image)*;
- **Флажок** - ячейка будет отображать значок флажка для булевых типов данных. Настройки ячейки аналогичны настройкам компонента *Флажок (Check)*;
- **Rtf текст** - ячейка будет отображаться как rich текст. Настройки ячейки аналогичны настройкам компонента *Rtf текст (RichText)*.

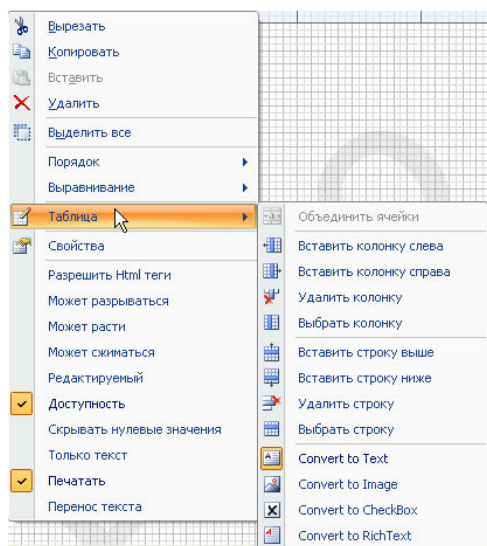
За вид ячейки отвечает свойство *Вид ячейки (CellType)*.



Кроме свойства *Вид ячейки (CellType)* для изменения вида ячейки можно использовать кнопку быстрого доступа ячейки.

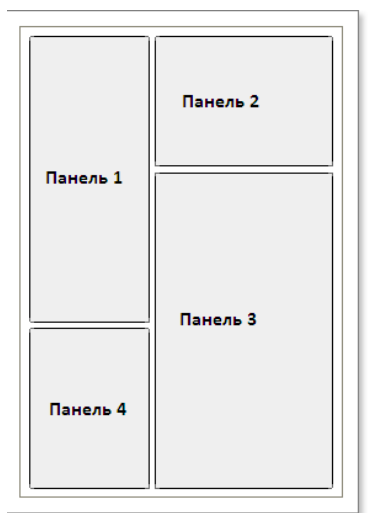


Или контекстное меню ячейки.



Контейнеры

Контейнер – это прямоугольная область, в которой могут содержаться другие компоненты, включая секции. При перемещении контейнера, вместе с ним перемещаются и все компоненты, которые находятся на нем. Контейнер может находиться, как на секции, так и на странице. Это дает уникальные возможности при создании отчетов.



Размещение секций в контейнере

Контейнер может быть размещен на странице, в другом контейнере, на секции. В контейнере могут находиться почти все компоненты отчета. Однако, в контейнере могут находиться не все секции. Ниже приведена таблица со списком секций и указанием мест, где эти секции могут быть размещены.

Наименование секции	Можно ли разместить секцию в контейнере
Заголовок отчета	Нет
Итог отчета	Нет
Заголовок страницы	Нет
Итог страницы	Нет
Заголовок группы	Да
Итог группы	Да
Данные	Да
Дочерний	Да
Заголовок данных	Да
Итог данных	Да

Как видно из таблицы, в контейнере не могут содержаться секции отчета и секции страницы. Все остальные секции могут быть расположены в контейнере и использованы при построении отчета.

Размещение контейнеров

В отчете могут встречаться три варианта размещения контейнера: *на странице*, *на секции* и *в другом контейнере*. Рассмотрим эти варианты использования контейнеров.

Первый вариант – **расположение контейнера на странице**. Основное предназначение – это организация нескольких независимых друг от друга потоков печати. При этом контейнеры могут находиться в произвольном месте страницы. Каждый контейнер представляет из себя как бы небольшую страницу. Т.е. контейнеры позволяют расположить на одной странице несколько небольших страниц со своими секциями и компонентами. Таким образом, можно построить множество сложных отчетов. Количество контейнеров, расположенных на одной странице, неограниченно.

Второй вариант – это когда **контейнер находится на секции**. Этот вариант имеет существенное ограничение. Контейнер не может содержать секций. В этом варианте контейнер может использоваться только для расположения компонентов для отображения информации.

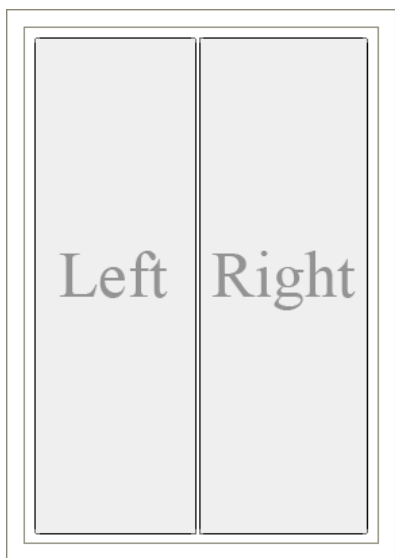
❖ **Важно!** Если контейнер расположен на секции, то он не может содержать секций.

Третий вариант – когда **один контейнер расположен в другом контейнере**. Этот вариант является смесью двух предыдущих. Т.е., если контейнер расположен в контейнере, который в свою очередь расположен не на секции, то контейнер может содержать секции. Если же он расположен не в контейнере, который в свою очередь расположен на секции, то в нем уже нельзя расположить секции.

Использование свойства **Печатать на** несколько отличается от использования в случае со страницами. При использовании этого свойства компонентами, расположенными в контейнере, формируется не новая страница. Управление передается следующему по порядку контейнеру. Если же это был последний по порядку контейнер, то тогда формируется новая страница.

Отчеты SIDE BY SIDE

Одним из типов отчетов, в котором контейнеры могут существенно ускорить создание отчетов, является *Side by Side*. В таком типе отчета параллельно выводятся на печать два списка строк. Причем оба списка абсолютно независимы друг от друга. Обычно для создания такого отчета необходимо использовать *Вложенный отчет* (иногда его называют суб-отчет). Генератор отчетов также имеет такую возможность. Однако, с использованием контейнеров, такой отчет строится быстро и просто.



Создадим отчет *Side by Side*. На странице располагаем два контейнера. У одного контейнера свойство *Тыковка* установим значение *К левому краю*. У второго в значение *К правому краю*. Это необходимо, для того чтобы контейнеры были пристыкованы к краям страницы и занимали по высоте все доступное место. Если в этом нет необходимости – этого можно не делать. Между контейнерами, в центре страницы оставим небольшое пространство, для разделения двух списков. Теперь в первом контейнере создадим две секции *Заголовок данных* и *Данные*. При помощи этих секций будем выводить первый список. Во втором контейнере сделаем то же самое, но для второго списка. В результате, после построения отчета, на странице будет выведено два списка параллельно.

Companies		Products	
Company	Phone	Product	Price
Hangry Crab & Right Cheese	20437562	igen Cheese	46.00p
Winkel Trading	11461555-8888	jack'n'New Original Cream Cheese	9.00p
Kinglich Cream	0555-09636	Korbu	6.00p
Luzernecheeondrom	305884-41	Loakbills	10.00p
Luximon wine	6037501-49	Laughing Lumberjack Lager	14.00p
Luzu Khouary Stem	05001555-369	Longale Tids	10.00p
Lufinone Markland	6848-604884	Louisiana Fatsy Fire Pepper Stum	21.00p
Luzu Stopy Stup	61451555-008	Louisiana Fire Spicatt Chen	17.00p
LUJA Supermarche	190581-808	Mangia o Great Apple	03.00p
LHU-Deekawee	1034-5692	Misurpene Faldit	32.00p
Lufinone Nye Rawsum	05001555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p
Megacard Almond Heart	005-64600	Misurpene Faldit	32.00p
Melton Cheay	1002011807	Misurpene Faldit	32.00p
Mere Palatou	05141555-008	Misurpene Faldit	32.00p

(c) 2009-2006 Stimulsoft Page 2 of 8

Существуют отчеты, в которых необходимо на одной странице **Вывести множество таблиц**. Причем таблицы выводятся в разных местах на странице. В большинстве случаев такой отчет строится с использованием *Вложенных отчетов*. Однако, построить такой отчет с использованием контейнеров не составляет особого труда. Все что необходимо сделать – расположить в нужных местах контейнеры, а в контейнерах расположить необходимые секции.

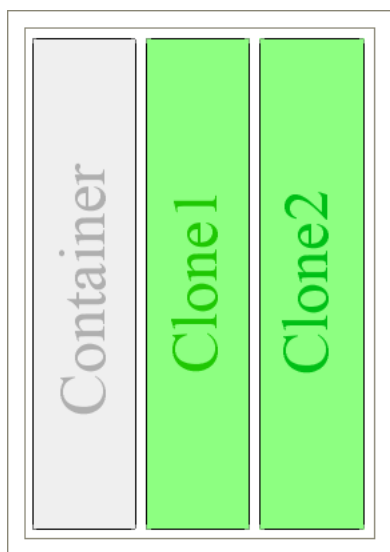
Table1	
1	
2	
3	
4	
5	
6	
7	
8	

Table2	
1	
2	
3	
4	
5	
6	
7	

Table3	
1	
2	
3	
4	
5	

Table4	
1	
2	
3	
4	
5	
6	

В состав компонентов генератора отчетов входит уникальный компонент – **Клон**. Этот компонент предназначен для клонирования части отчета в определенное место отчета. Клонировать можно только содержимое контейнера. Процесс клонирования происходит следующим образом. Разместим на странице контейнер. В контейнере разместим секции для вывода списка. Контейнер разместим в левой части страницы. В правой части страницы разместим компонент **Клон**. Далее в дизайнера компонента **Клон** указываем, какой контейнер необходимо клонировать. В нашем случае указываем контейнер, который создали ранее на странице.



Запустим отчет на выполнение. Сначала был построен контейнер. Был выведен список в левой части страницы. После чего, вывод списка был продолжен в правой части страницы, в том месте, где расположен компонент **Клон**. Это произошло потому, что компонент **Клон** клонировал все секции, которые находились в контейнере. Используя компонент **Клон** можно строить очень сложные отчеты с колонками. Первая колонка выводится при помощи контейнера, последующие при помощи компонентов **Клон**. При использовании клонирования, следует учитывать порядок расположения компонентов **Клон** на странице. При построении отчета первым всегда выводится контейнер. После него происходит построение компонентов **Клон**.

Использование авторазмера

Компоненты могут автоматически изменять свой размер в зависимости от информации, которую они содержат. Выводимый текст может не уместиться в размерах компонента. Или размер изображения меньше размеров компонента, который выводит это изображение. Ча-

ще всего автоматическое изменение размеров компонентов применяется тогда, когда необходимо вывести текст неизвестного заранее объема в компоненте. Можно конечно заранее попробовать подобрать размер компонента таким образом, чтобы текст вмещался во всех случаях, но скорее всего в данном случае пространство страницы будет использоваться не эффективно. Если объем выводимого текста в одном компоненте велик, к примеру, на половину страницы, то для рационального использования места на странице можно применять автоматическое изменение размеров.

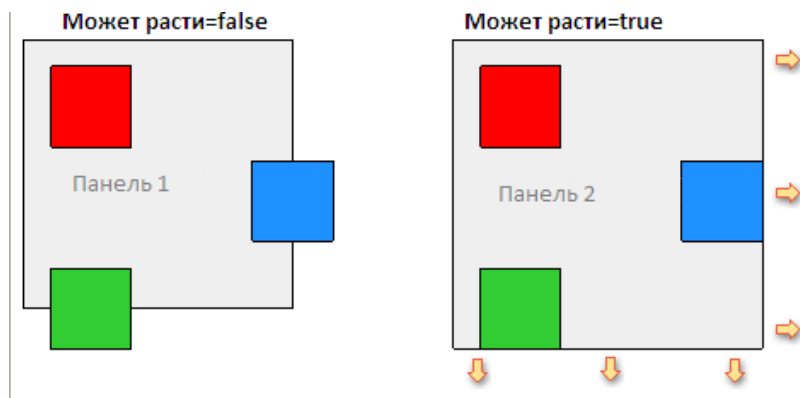
Автоматическим изменением размеров

Автоматическим изменением размеров можно управлять при помощи двух свойств – *Может расти* и *Может сжиматься*. Если свойство *Может расти* установлено в *Да*, это означает, что компонент может увеличить свой размер, если содержащаяся в нем информация не помещается в размерах компонента. Например, не уместится текст, не видно полностью изображения. Если свойство *Может сжиматься* установлено в *Да*, то компонент может уменьшить свой размер так, чтобы его размеры совпадали с размерами информации содержащейся в нем. Таким образом можно избежать лишних затрат места.

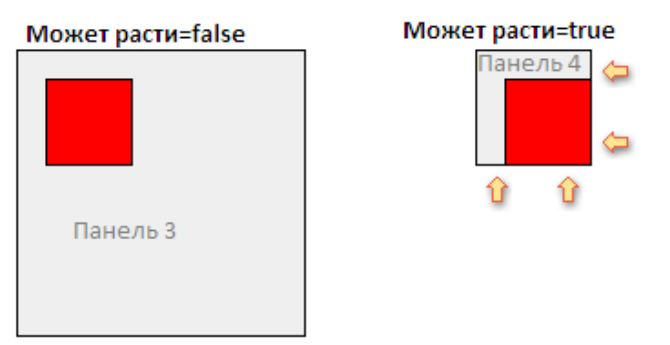
Генератор отчетов допускает одновременную установку в *Да* свойств *Может расти* и *Может сжиматься*. В этом случае если требуется увеличить размеры – размеры будут увеличены. Если требуется уменьшить размеры, то будут уменьшены.

Автоматическое изменение размеров контейнеров

Контейнеры не могут содержать какую то визуальную информацию для отображения в отчета. Соответственно может показаться, что свойства *Может расти* и *Может сжиматься* их не касаются. Однако это не так. Контейнеры могут содержать другие компоненты, которые имеют определенные размеры и позицию. В зависимости от этих параметров компонентов находящихся в контейнере и изменяются размеры контейнера. Предположим, что в контейнере находятся компоненты, положение и размеры которых выходят за размеры контейнера. Свойство *Может расти* установлено в *Да*. В этом случае размеры контейнера будут увеличены так, чтобы все компоненты находящиеся в этом контейнере находились в пределах его размеров.

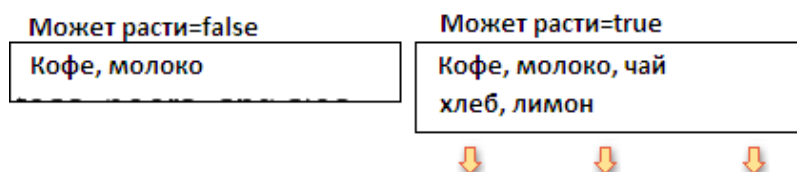


В случае если свойство *Может сжиматься* установлено в *Да* и размеры общей области всех компонентов меньше чем размеры контейнера, размеры контейнера будут уменьшены до размеров общей области всех компонентов.

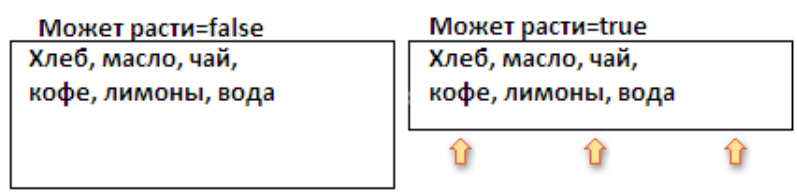


Автоматическое изменение размеров текста

Автоматическое изменение размеров текстовых компонентов отличается от других компонентов. Свойство *Может расти* и *Может сжиматься* влияют только на высоту текстового компонента. К примеру, если текст не уместился в пределах размера компонента, то высота будет увеличена.



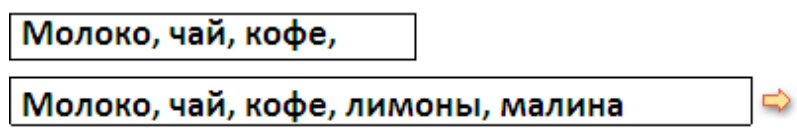
Если наоборот есть лишнее место, то генератор отчетов уменьшит высоту текстового компонента.



Также как и в случае с другими компонентами допускается устанавливать одновременно два свойства *Может расти* и *Может сжиматься* в *Да*. В этой ситуации высота будет уменьшена или увеличена в зависимости от выводимого текста.

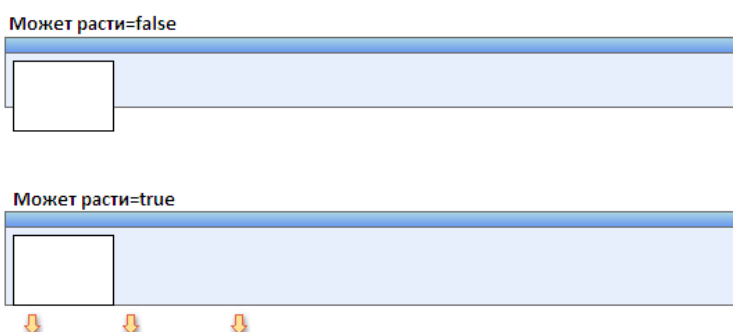
На изменение размеров кроме свойств *Может расти* и *Может сжиматься* влияет также специальное **свойство Автоширина**. Если свойство *Автоширина* установлено в *Да*, то текстовый компонент будет изменять свою ширину в зависимости от ширины выводимого текста. Свойства *Может расти*, *Может сжиматься* и *Автоширина* могут использоваться одновременно.

- ❖ **Важно!** Если свойство *Перенос текста* установлено в *Нет*, то высота текста зависит от установки свойств *Может расти* и *Может сжиматься*. Если свойство *Перенос текста* равно *Нет*, то будет изменяться ширина текста.



Автоматическое изменение размеров секций

Секции являются контейнерами, поэтому их поведение при автоматическом изменении размеров во многом повторяет поведение контейнеров. Размеры секции изменяются в зависимости от компонентов находящихся на секции. Однако, у секций есть одна особенность. Секции могут автоматически изменять только высоту (ширину для кросс-секций). К примеру, если на секции расположен компонент, у которого нижняя граница выходит за пределы нижней границы секции. В этом случае, если установить секции свойство *Может расти* равным *Да*, высота секции будет увеличена генератором отчетов. После увеличения, нижняя граница секции будет совпадать с нижней границей компонента. Т.е. область компонента полностью находится в области секции.



Соответственно, если установлено свойство *Может сжиматься* в *Да*, и есть свободное место между нижней границей компонента и нижней границей секции – высота секции будет уменьшена.

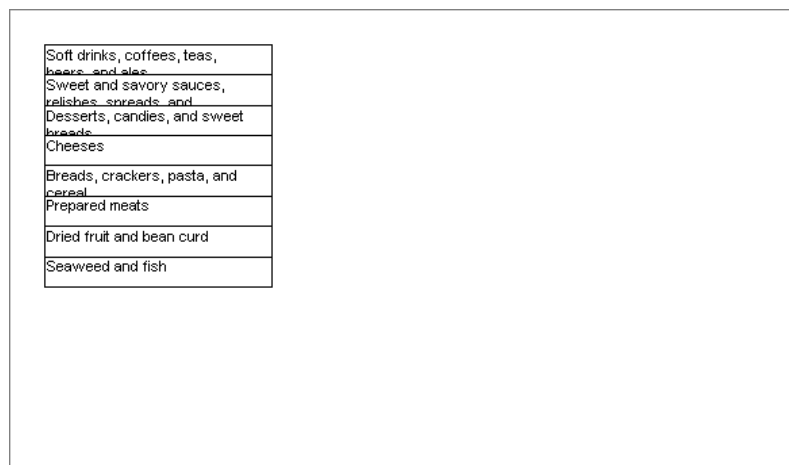


Совместное изменение размеров секции и компонентов

Наиболее частая ситуация совместного изменения размеров компонентов это изменение размеров компонента *Текст* и секции *Данные*, на которой находится этот компонент *Текст*. Рассмотрим пример. Предположим, что необходимо вывести текст, причем размер текста заранее неизвестен. Для этого, на странице создадим секцию *Данные*, на которой создадим компонент *Текст*, который будет выводить текст. У текстового компонента установим свойство *Перенос* текста в *Да*, чтобы он выводил текст в несколько строк.



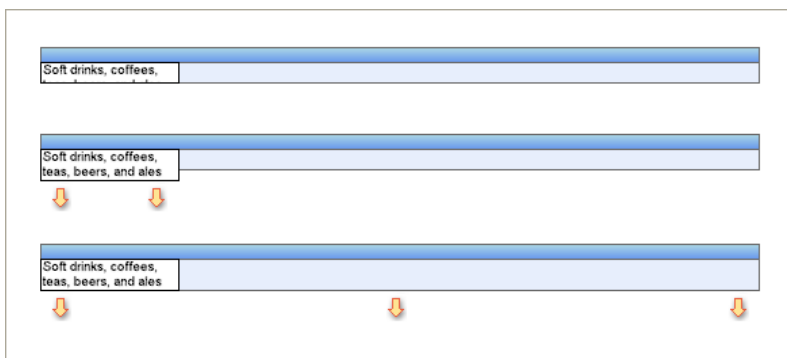
Также установим свойство *Может расти* в *Да*, для того чтобы компонент *Текст* мог увеличить свою высоту, если его размеров не хватит чтобы полностью отобразить весь текст. Если после этого запустить отчет на выполнение можно увидеть, что высота текстового компонента меняется, но не изменяется высота секции *Данные*. В итоге текстовые компоненты наезжают друга на друга.



Для того чтобы секция *Данные* изменила свою высоту так, чтобы все компоненты находящиеся на ней полностью были отображены, необходимо установить свойство *Может расти* секции *Данные* также в *Да*. После этого секция *Данные* будет изменять свои размеры в зависимости от размеров компонента *Текст*.

Soft drinks, coffees, teas, beers, and ales
Sweet and savory sauces, relishes, spreads, and seasonings
Desserts, candies, and sweet breads
Cheeses
Breads, crackers, pasta, and cereal
Prepared meats
Dried fruit and bean curd
Seaweed and fish

Работа генератора отчётов происходит следующим образом. Сначала происходит расчет размеров компонента *Текст*. Если есть необходимость, они увеличиваются или уменьшаются. В нашем случае была увеличена высота. После этого секция *Данные* просматривает все компоненты, которые находятся на ней. Если встречаются компоненты, которые выходят за границы секции, то размеры секции увеличиваются.



Привязывание нижней границы компонента

В предыдущей статье описывался пример с совместным изменением размеров секции и текстового компонента, находящегося на этой секции. В этом примере изменял свои размеры только один компонент. Но что делать, если на секции находятся несколько компонентов. Одна из них изменяет размеры, а другие нет.

ДатаВанд1; Источник данных: Categories		
{Categories.Category}	{Categories.Description}	{Categories.CategoryName}

В итоге, если посмотреть на построенный отчет, то будут видны промежутки.

1	Soft drinks, coffees, teas, beers, and ales	Beverages
2	Sweet and savory sauces, relishes, spreads, and seasonings	Condiments
3	Desserts, candies, and sweet breads	Confections
4	Cheeses	Dairy Products
5	Breads, crackers, pasta, and cereal	Grains/Cereals
6	Prepared meats	Meat/Poultry
7	Dried fruit and bean curd	Produce
8	Seaweed and fish	Seafood

Для изменения высоты компонент не изменяющих можно привязать нижнюю границу компонента к нижней границе секции, на которой находится этот компонент. Для этого используется свойство *Расту в высоту*. По умолчанию свойство установлено в *Нет*. Установив значения этого свойства в *Да*, для всех компонентов, которые не изменяют свои размеры, можно заставить их нижние границы прилипнуть к нижней границе секции. Свойство *Расту в высоту* привязывает нижнюю границу компонента не только к нижней границе секции, но и к нижней границе контейнера. Свойство *Расту в высоту* можно устанавливать и для компонентов, которые автоматически изменяют свои размеры. В этом случае, если после изменения размеров компонента, его нижняя граница не совпадает с нижней границей секции, на которой он находится, размеры компонента будут скорректированы. После этой операции отчет принимает нормальный вид.

1	Soft drinks, coffees, teas, beers, and ales	Beverages
2	Sweet and savory sauces, relishes, spreads, and seasonings	Condiments
3	Desserts, candies, and sweet breads	Confections
4	Cheeses	Dairy Products
5	Breads, crackers, pasta, and cereal	Grains/Cereals
6	Prepared meats	Meat/Poultry
7	Dried fruit and bean curd	Produce
8	Seaweed and fish	Seafood

Если на одной секции находятся сразу несколько компонентов, которые могут изменять свои размеры, то можно установить значение свойства *Расту в высоту* для всех этих компонентов в *Да*. В результате, высоты этих компонентов всегда будут корректироваться в зависимости от высоты компонента, который имеет наибольшую высоту.

Смещение компонентов

Если один компонент увеличит свою высоту, то он перекроет частично второй компонент. Для предотвращения этого используется свойство **Режим смещения**. По умолчанию это свойство установлено в *При увеличении размера*. Если у компонента изменяется высота, то все компоненты, верхняя граница которых находится ниже или равна верхней границе изменяемого компонента, будут смещены вниз на величину изменения высоты. При этом не компоненты могут не пересекаться. Уменьшение высоты компонента не ведет к изменению положения компонент, которые находятся под ним.

Свойство *Режим смещения* работает только при увеличении высоты. К примеру, на секции *Данные* расположим пять компонентов. Один из этих компонентов (далее изменяемый компонент) будет автоматически увеличивать свою высоту в зависимости от текста, который в нем находится. Остальные четыре компонента будут изменять или не изменять свое положение в зависимости от этих изменений. Рассмотрим последовательно поведение других четырех компонентов.

DataBand1; Источник данных: Categories	
{Categories.Description	Component3
Component1	Component2
Component4	

У *Component1* верхняя граница совпадает с нижней границей изменяемого компонента. Поэтому он будет менять свое положение по вертикали в зависимости от высоты изменяемого компонента. *Component2* по горизонтали не находится под изменяемым компонентом, но его верхняя граница совпадает с нижней границей изменяемого компонента. Положение *Component2* будет изменяться. *Component3* по горизонтали не находится под изменяемым компонентом. Верхняя граница совпадает с верхней границей изменяемого компонента. Однако его верхняя граница гораздо выше нижней границы изменяемого компонента, поэтому его положение останется без изменений. У *Component4* верхняя граница находится ниже нижней границы изменяемого компонента, поэтому его положение будет изменено.

Soft drinks, coffees, teas, beers, and ales	Component3
Component1	Component2
Component4	
Sweet and savory sauces, relishes, spreads, and seasonings	Component3
Component1	Component2
Component4	
Desserts, candies, and sweet breads	Component3
Component1	Component2
Component4	
Cheeses	Component3
Component1	Component2
Component4	
Breads, crackers, pasta, and cereal	Component3
Component1	Component2
Component4	
Prepared meats	Component3
Component1	Component2
Component4	

В данном примере у всех компонентов было включено свойство *Режим смещения*. Если бы это свойство было выключено, компоненты не изменили бы своего положения.

Расчет итогов

Во многих отчетах необходимо рассчитывать итоги – суммы по странице, количество строк в группе, среднее значение и т.д. Для этих целей в генераторе отчетов используются агрегатные функции. При помощи агрегатных функций можно произвести расчет сумм, количества строк, средних значений, максимальных значений, минимальных значений, получить первые значения и списка, получить последние значения из списка, по определенному диапазону строку. Диапазоном строк могут быть все строки списка, строки на одной странице, строки из одной группы, строки из одного контейнера и т.д.

Генератор отчетов поддерживает семь агрегатных функций:

- *Sum* - возвращает сумму заданного выражения;
- *Count* - возвращает количество строк в заданном диапазоне;
- *Avg* - возвращает среднее значение заданного выражения;
- *Min* - возвращает минимальное значение заданного выражения;
- *Max* - возвращает максимальное значение заданного значения;
- *First* - возвращает первое значение в заданном диапазоне;
- *Last* - возвращает последнее значение в заданном диапазоне.

Методика расчета итогов

Для того чтобы рассчитать значение агрегатной функции генератор отчетов использует события компонентов. Для расчета ему необходимы три события у компонента: начало построения (инициализация агрегатной функции), построение одного элемента (обработка одной строки) и конец построения (вывод результатов расчета).

Обычно генератор отчетов использует события *Начало построения*, *Построение* и *Конец построения* секции *Данные* для проведения всех расчетов. Но могут быть и другие ситуации. К примеру, для того чтобы рассчитывать итоги по группе, генератор отчетов использует события *Начало построения*, *Построение* и *Конец построения* у секции *Заголовок группы*, а не у секции *Данные*. Также возможно и комбинирование, например, инициализация и расчет в событиях секции *Данные*, а вывод получившихся данных в событии *Конец построения* отчета. Такая схема расчета придает большую гибкость при расчете агрегатных функций в генераторе отчетов и позволяет произвести все расчеты в один проход.

Вывод итогов в любом месте отчета

Обычно компоненты, в текстовом выражении которых указан вызов агрегатной функции должны быть расположены на итоговой секции или на секции *Данные*. Если же требуется вывести итог например в секции *Заголовок данных*, то это вообще невозможно или возможно только при помощи скрипта. Однако, это правило не относится к генератору отчетов. Компонент с агрегатной функцией может быть расположен на любой секции. Компоненты с агрегатными функциями можно размещать в любом месте отчета. Также допускается размещение и просто на странице и на других страницах отчета.

Возможно, рассчитывать сумму значений по списку и вывести его в заголовке списка. Или рассчитать количество строк в списке и вывести это значение в начале страницы. При этом есть небольшое ограничение. Необходимо точно указывать к какой секции *Данные* относится агрегатная функция. Для этого необходимо дополнительно указать секцию *Данные* в виде аргумента функции. К примеру: `{Sum(Данные1, Products.UnitsInStock)}` - это выражение вернет сумму значений колонки *Products.UnitsInStock* для каждой строки секции *Данные1*. В случае агрегатной функции *Count*: `{Count(Данные1)}` - это выражение вернет количество строк секции - *Данные1*.

Тип результата агрегатной функции

По умолчанию все агрегатные функции возвращают значение типа *Decimal* (за исключением функции *Count*). Генератор отчетов позволяет производить расчеты дополнительно с использованием двух типов данных - *Double* и *Int64*. Для того чтобы агрегатная функция возвращала результат расчета, используя тип данных *Double*, добавьте латинскую букву *D* в верхнем регистре к названию функции. Для расчетов с использованием типа *Int64* необходимо добавить латинскую букву *I* в верхнем регистре. Например, для расчета суммы с использованием типа *Int64* необходимо написать: `SumI`. Для типа *Double*: `SumD`. Для типа *Decimal*: `Sum`. Такое разделение позволит избежать потерь при расчетах агрегатных функций.

Написание агрегатной функции	Тип возвращаемого значения
Sum	Decimal
SumD	Double
SumI	Int64
Count	Int64

Функция *Count* всегда рассчитывается с использованием типа *Int64*.

Выражения с агрегатными функциями

Выражения с использованием агрегатных функций можно использовать только в выражении *Text* компонента *Текст*. Для того чтобы просуммировать все значения одного столбца достаточно написать в текстовом выражении компонента: `{Sum(DataSource.Column)}`. Также допускается использование сложных выражений: `{100 + Sum(DataSource.Column) * 2}`.

- ❖ **Важно!** Не допускается использование одновременно более одной агрегатной функции в текстовом выражении. В этом случае генератор отчетов не сможет корректно вывести результаты расчетов. Если все же такая необходимость имеется, то рекомендуется использовать в этом случае события отчета для расчета итогов.

К примеру, необходимо вывести количество строк и сумму значений выражения. Для этого необходимо добавить к отчету секцию *Итог данных*. На этой секции размещаем два компонента *Текст*. В первом компоненте в текстовом выражении указываем: `{Count()}` - в этом компоненте будет выведено количество строк. Во втором: `{Sum(Products.UnitsInStock)}` - в этом компоненте будет выведена сумма значений из столбца *UnitsInStock*.

HeaderBand1	
Product name	Units in stock
DataBand1; Data Source: Products	
{Products.ProductName}	{Products.UnitsInStock}
FooterBand1	
{Count()}	{Sum(Products.UnitsInStock)}

Как видно из примера для расчета количества строк функции *Count* не нужно дополнительных аргументов. Функции *Sum* был задан только один аргумент – выражение, которое будет должно быть просуммировано. Т.е. генератор отчетов сам определил, к какой секции Данные относятся эти агрегатные функции и сколько раз необходимо вызвать эти функции.

Product name	Units in stock
Alice Mutton	0
Aniseed Syrup	13
Boston Crab Meat	123
Camembert Pierrot	19
Camarvon Tigers	42
5	197

Это произошло, потому что текстовые компоненты, в которых были использованы агрегатные функции, были размещены на итоговой секции *Итог данных*. Эта секция всегда относится, к какой то секции *Данные*. Что в свою очередь дает возможность генератору отчетов автоматически связать агрегатную функцию и конкретную секцию *Данные*. В генераторе отчетов есть несколько видов итоговых секций. К итоговым секциям относятся:

- *Итог отчета* - секция используется для вывода итогов по всему отчету;
- *Итог страницы* - секция используется для вывода итогов по странице;
- *Итог данных* - секция используется для вывода итогов по списку;
- *Итог группы* - секция используется для вывода итогов по группе.

Расположение компонентов с агрегатными функциями на любой из указанных выше секций позволяет генератору отчетов точно определить к какой секции *Данные* относится агрегатная функция. Дополнительно к указанным секциям компонент с агрегатной функцией можно разместить и на самой секции *Данные*. В этом случае, на каждой строке данных будет выведен результат расчета агрегатной функции по всем строкам.

- ❖ **Важно!** В большинстве случаев генератор отчетов автоматически связывает агрегатную функцию и секцию *Данные*.

Итоги с условием

Часто возникает задача рассчитать итоги с условием. К примеру, необходимо сложить все значения больше нуля. Генератор отчетов позволяет добавлять к агрегатной функции условие. Для добавления условия к названию агрегатной функции необходимо добавить слово *If* латинскими буквами, и дополнительный аргумент с условием. К примеру: $\{SumIf(Изделия.Запас, Изделия.Запас > 0)\}$ - это выражение вернет сумму элементов, причем каждый элемент должен быть больше нуля. Для функции *Count*: $\{CountIf(Изделия.Запас == 0)\}$ - это выражение вернет количество элементов равных нулю. Если необходимо произвести расчет с использованием типа *Double* или *Int64*, то сначала добавляем латинскую букву *D* или *I*, а затем слово *If*. К примеру: $\{SumDIf(Изделия.Запас, Изделия.Запас > 0)\}$.

Расчет итогов по странице и по контейнеру

Расчитать итог по странице или по контейнеру можно следующим образом. Для этого необходимо добавить спереди к названию агрегатной функции латинскую букву *c* в нижнем регистре. К примеру: $\{cCount(DataBand1)\}$ - выражение вернет количество строк на одной странице. *Страница* – это тоже контейнер, поэтому расчеты итогов по ней идет по такому же принципу, как и для контейнера.

При расчете итогов по контейнеру или по странице, желательно указывать секцию *Данные*, по которой производится расчет агрегатной функции. Это необходимо, потому что на одной странице может находиться более одной секции *Данные*. Для расчета агрегатной функции по контейнеру достаточно добавить латинскую букву *c* перед названием агрегатной функции. На одной странице или контейнере может использоваться любое количество агрегатных функций. Генератор отчетов не налагает никаких ограничений на это. Допускается комбинирование итогов по странице с условием. К примеру: $\{CountIf(DataBand1, Products.UnitsInStock = 0)\}$ - выражение вернет количество элементов на этой странице равных нулю.

Расчет итогов по колонке

С помощью генератора отчетов можно рассчитать итоги по колонке. Спереди к наименованию агрегатной функции добавляется строка *col* (от слова *column*) в нижнем регистре. К примеру: `{colCount()}` - выражение вернет количество строк в одной колонке.

22. Familia Anquihallo	(11) 555-9857	66. Raltesnake Canyon Grocery	(506) 555-9939
23. FBSA Fabrica Inter. Salchichas SA	(91) 555 94 44	67. Reggiani Caseifici	0522-596721
24. Folies gourmandes	20.16.10.16	68. Ricardo Adocicados	(21) 555-3412
25. Folk och de H B	0695-34 67 21	69. Richter Supermarki	0897-034214
26. France res lauraton	40.32.21.21	70. Romero y Romillo	(91) 7 45 6200
27. Franchi S.p.A.	011-4988260	71. Sanit Gourme I	07-98 92 35
28. Frankenwiesand	089-0877310	72. Saue-e-oi Marke Is	(208) 555-8097
29. Frita Bacalhau: Frutos do Mar	(1) 354-2534	73. Seven Seas Imports	(171) 555-1717
30. Galetedel gas Inonomo	(93) 203 4560	74. Simons bisio	31 12 34 96
31. Gostos Coctna Tpicos	(95) 555 82 82	75. Spécialités dumonde	(1) 47.55.60.10
32. Gourme I Lanchone Is	(11) 555-9482	76. Spill I Rali Beer&Ale	(307) 555-4680
33. Great Lakes Food Marke I	(503) 555-7555	77. Suprêmes d'élites	(071) 23 67 22 20
34. GROSSELLA-Res laurante	(2) 283-2951	78. The Big Cheese	(503) 555-3612
35. Harard Games	(21) 555-0081	79. The Cracker Box	(406) 555-5834
36. HILARIO II-Abas los	(5) 555-1340	80. Toms Spedal Ibin	0251-031259
37. Hungry Coyote Import Store	(503) 555-6874	81. Torluga Res laurante	(5) 555-2933
38. Hungry Owl All-Right Grocers	2967 542	82. Tradiglo Hipemercados	(11) 555-2167
39. Iriani Tradiing	(198) 555-8888	83. Trail's Head Gourme I Provisions	(206) 555-8257
40. Jönlich Essen	0555-0876	84. Vantelime I	86 21 32 43
41. La come d'abondance	30.59.84.10	85. Victualites ens lod	78.32.54.86
42. Lemisond'Asie	61.77.61.10	86. Vins e lalcoods Cheveller	26.47.15.10
43. Laughing Bacchus Wine Cellars	(504) 555-3382	87. Wlartan Herku	981-443655
44. Lazy K Kountry Store	(509) 555-7969	88. Welling ton Importadca	(14) 555-6122

Company	Phone
89. Nili le Cloue Marke Is	(206) 555-4112
90. Nlman Kala	90-224 8858
91. Nolski Zalad	(26) 642-7012

Расчет итогов по колонке в генераторе отчетов имеет одно ограничение. Итоги можно рассчитывать только по колонкам на странице. Рассчитывать итоги по колонкам на секции *Данные* не допускается.

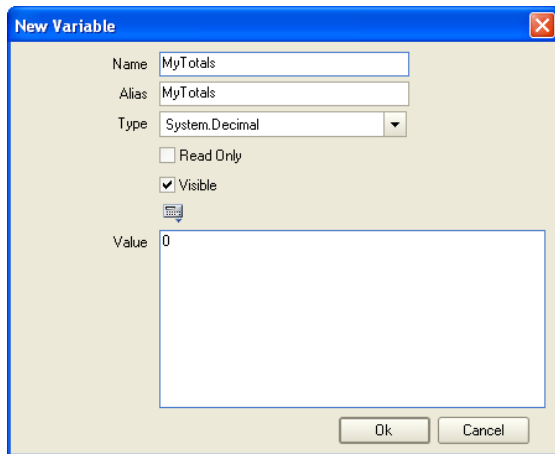
- ❖ **Важно!** *Допускается рассчитывать итоги только по колонкам на странице. Расчет итогов по колонкам расположенным на секции Данные невозможен.*

При расчете итогов по колонке желательно, текстовый компонент с агрегатной функцией размещать на секциях *Заголовок колонки*, *Итог колонки*, *Заголовок данных* или *Итог данных*. Для расчета агрегатной функции по колонке достаточно добавить строку *col* перед названием агрегатной функции. Можно рассчитывать неограниченное количество итогов по колонке. Каких либо ограничений не накладывается. Также допускается комбинирование итогов по колонке с условием. К примеру: `{colCountIf(Данные1, Products.UnitsInStock = 0)}` - выражение вернет количество строк в колонке у которых условие верно.

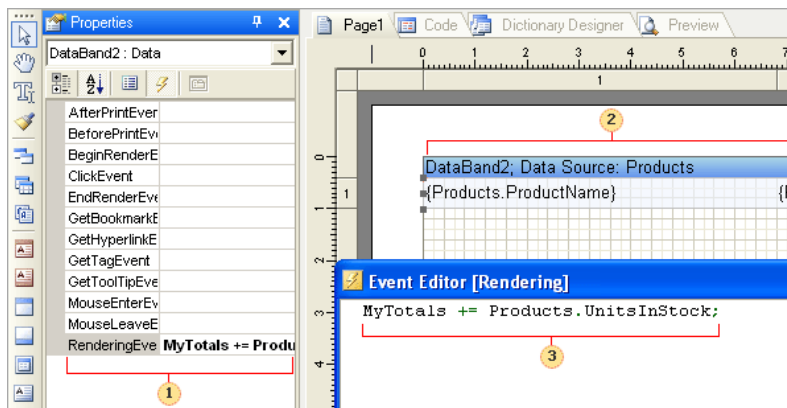
Расчет итогов в коде событий отчета

При помощи генератора отчетов также возможно произвести расчет агрегатных функций в коде событий отчета. Это позволяет вычислить агрегатные функции со сложной логикой или со сложным условием. Еще одним преимуществом является-то, что в ходе расчета к рассчитываемому значению можно обращаться из кода, влиять на ход расчета. Для того чтобы произвести такой расчет требуется переменная, которая будет хранить значение агрегатной функции. Для этого необходимо создать новую переменную в словаре данных.

- ❖ **Важно!** *Не допускается использование для хранения результата расчета агрегатных функций переменных объявленных в коде. Необходимо использовать переменные созданные в словаре.*

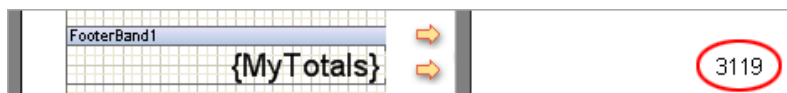


Указывается тип переменной, к примеру, *Decimal*, и первоначальное значение, к примеру, *0*. Далее у секции *Данные* в событии *Построение* заносится код для приращения переменной. К примеру, если необходимо посчитать сумму значений по полю *Products.ItemsInStock* код будет такой: `MyTotals += Products.ItemsInStock;`



Цифрой один на рисунке выше обозначено событие *Построение*. Цифрой два обозначена секция *Данные*, в событии *Построение*, которого будет происходить расчет. Цифра три - код для вычисления суммы.

Для обращения к переменной содержащей значение итога пишем в текстовом выражении: `{MyTotals}`. Если сразу после этих действий запустить отчет на выполнение, то, как только генератор отчетов дойдет до компонента, который содержит выражение с итоговой переменной, он сразу выведет этот компонент. Поэтому необходимо дать указание генератору отчетов, что необходимо вывести компонент только после построения всего отчета – когда переменная будет рассчитана до конца. Для этого необходимо у текстового компонента установить свойство *Обработать в конце* в *Да*. Выражения текстовых компонентов, у которых свойство *Обработать в конце* установлено в *Да*, всегда вычисляются в конце построения отчета. После приведенных действий итог будет рассчитан и выведен в указанном месте.



Расчет нарастающего итога

Нарастающий итог рассчитывается по такому же принципу, как и итоги при помощи кода в событиях. Единственное отличие в том, что для компонентов *Текст* не нужно устанавливать свойство *Обработать в конце* в *Да*. В этом случае, когда генератор отчетов будет обрабатывать компонент, который выводит на печать переменную (использующуюся для расчета), он будет выводить этот компонент немедленно, не дожидаясь конца построения отчета. Таким образом, если компонент *Текст* расположен на секции *Итог страницы*, то будет выведен итог по странице, а т.к. переинициализации не происходит, то переменная будет накапливать итог.

Невидимые секции

Многие отчеты использует скрытие секций по определенному условию. По умолчанию, генератор отчетов не будет учитывать выключенные секции *Данные*. Но если есть необходимость при расчете итога учитывать и невидимые секции *Данные*, то необходимо установить у секции *Данные* свойство *Вычислять невидимые* в *Да*. В этом случае, на печать будут выводиться только видимые секции *Данные*, а при расчете агрегатной функции будут учитываться все строки.

Итоги и автоматическое изменение размеров

Есть одна особенность при использовании автоматического изменения размеров компонента, который выводит результат расчета итога. Как правило, в тот момент, когда определяется размер компонента, результат расчета агрегатной функции еще неизвестен. Поэтому компонент не может корректно скорректировать свой размер с учетом результата агрегатной функции. Эту особенность необходимо учитывать при построении отчетов.

Синтаксис агрегатных функций

- Sum(expression);
- Sum(band, expression);
- SumIf(band, expression, condition);
- *expression* – выражение для расчета;
- *band* – название секции для выполнения расчета;
- *condition* – условие включения выражения в расчет.

В случае с расчетами по странице или контейнеру, синтаксис тот же за исключением добавление к написанию латинской буквы *c*:

- cSum(expression);
- cSum(band, expression);
- cSumIf(band, expression, condition).

Для расчета итогов по колонке, к наименованию агрегатной функции добавляется спереди строка *col*:

- colSum(expression);
- colSum(band, expression);
- colSumIf(band, expression, condition).

Функция *Count* отличается от остальных агрегатных функций тем, что она не имеет выражения для расчета. Синтаксис для этой функции приведен ниже.

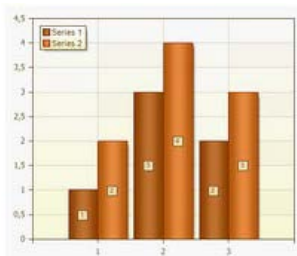
- Count();
- CountIf (condition);
- Count (band);
- CountIf(band, condition);
- cCount ();
- cCount (band);
- cCountIf(band, condition);
- colCount ();
- colCount (band);
- colCountIf(band, condition).

❖ **Важно!** При использовании языка программирования C#, все агрегатные функции нужно писать строго с соблюдением регистра.

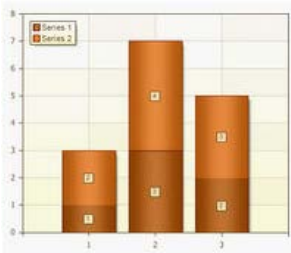
Диаграммы

Гистограммы (*Column Area*). Гистограммы используются для демонстрации изменений данных за определенный период времени или для иллюстрирования сравнения объектов. В гистограммах категории формируются по горизонтальной оси, а значения — по вертикальной. *Гистограммы* содержат следующие подтипы диаграмм:

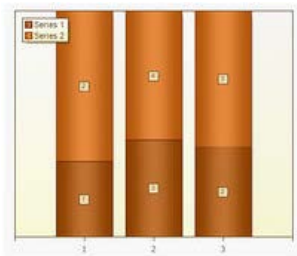
- **Сгруппированная гистограмма (Clustered Column).** В сгруппированных гистограммах значения сравниваются по категориям.



- **Гистограммы с накоплением (Stacked Column).** Диаграммы с накоплением показывают отношение отдельных частей к целому, позволяя сравнить вклад каждого значения в итог внутри категории. В гистограммах с накоплением значения выводятся в виде плоских вертикальных прямоугольников с накоплением.

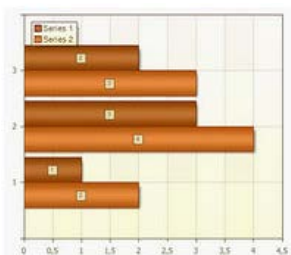


- **Нормированная гистограмма с накоплением (Full-Stacked Column).** Этот тип гистограмм позволяет сравнить процентную долю каждого значения с итогом внутри категории. В нормированных гистограммах с накоплением значения выводятся в виде плоских вертикальных нормированных прямоугольников с накоплением. Нормированные гистограммы с накоплением можно использовать при наличии двух или более рядов данных, если требуется обратить внимание на их вклад в итоги, особенно если итоги одинаковы для каждой категории.

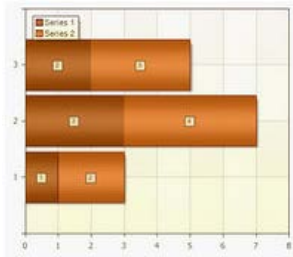


Линейчатые диаграммы (Bar Area). Данные, которые расположены в столбцах или строках, можно изобразить в виде линейчатой диаграммы. Линейчатые диаграммы иллюстрируют сравнение отдельных элементов. Линейчатые диаграммы рекомендуется использовать, если *Метки осей* имеют большую длину или выводимые значения представляют собой длительности.

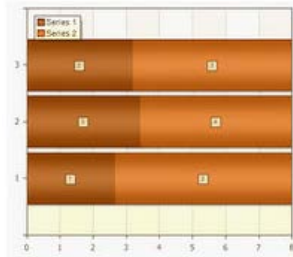
- **Сгруппированные линейчатые диаграммы (Clustered Bar).** В сгруппированных линейчатых диаграммах значения сравниваются по категориям. В сгруппированной линейчатой диаграмме категории располагаются вдоль вертикальной оси, а значения — вдоль горизонтальной оси.



- **Линейчатые диаграммы с накоплением (Stacked Bar).** Линейчатые диаграммы с накоплением отображают отношение отдельных элементов к общей величине.



- **Нормированная линейчатая диаграмма с накоплением (Full-Stacked Bar).** Этот тип линейчатых диаграмм позволяет сравнить процентную долю каждого значения с итогом внутри категории.

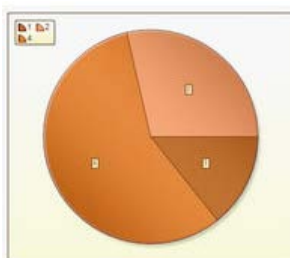


Круговые диаграммы (Pie Area). Данные, которые расположены в одном столбце или строке, можно изобразить в виде круговой диаграммы. Круговая диаграмма демонстрирует размер элементов одного ряда данных пропорционально сумме элементов. Точки данных на круговой диаграмме выводятся в виде процентов от всего круга. Круговые диаграммы отображают долю каждого значения в общей сумме в плоском виде. Круговые диаграммы рекомендуется использовать, если:

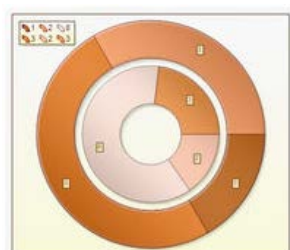
- Требуется отобразить только один ряд данных;
- Все значения, которые требуется отобразить, неотрицательны;
- Почти все значения, которые требуется отобразить, больше нуля;
- Значения принадлежат не более чем семи категориям;
- Категории соответствуют частям общего круга.

Виды диаграмм.

- **Круговые диаграммы (Pie).** Круговые диаграммы отображают долю каждого значения в общей сумме в плоском и объемном виде. Можно вручную отделить сектора круговой диаграммы, чтобы подчеркнуть представляемые ими значения.

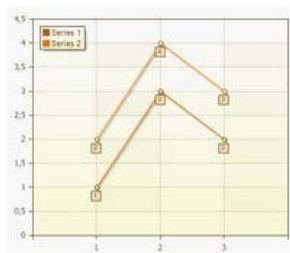


- **Кольцевые диаграммы (Doughnut).** Данные, которые расположены только в столбцах или строках, можно изобразить в виде кольцевой диаграммы. Как и круговая диаграмма, кольцевая диаграмма отображает отношение частей к целому, но может содержать более одного ряда данных. В кольцевых диаграммах данные выводятся в виде колец, при этом каждое кольцо соответствует одному ряду данных.

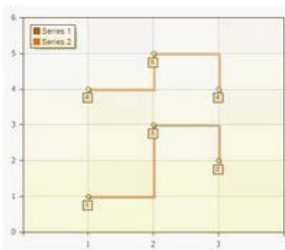


Линии (Line Area). Данные, которые расположены в столбцах или строках, можно изобразить в виде линий. Линии позволяют изображать непрерывное изменение данных с течением времени в едином масштабе; таким образом, они идеально подходят для изображения трендов изменения данных с равными интервалами. На графиках категории данных равномерно распределены вдоль горизонтальной оси, а значения равномерно распределены вдоль вертикальной оси.

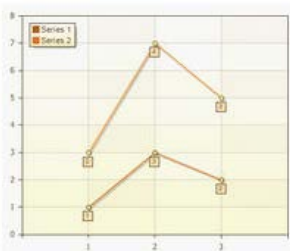
- **Линии (Line).** Линии, выводимые с маркерами, указывающими на отдельные значения данных, или без них, используются для демонстрации непрерывного изменения трендов со временем или в зависимости от упорядоченных категорий, особенно при наличии большого количества точек данных, если важен порядок их вывода. Линии без маркеров рекомендуется использовать при аппроксимации множества категорий значений.



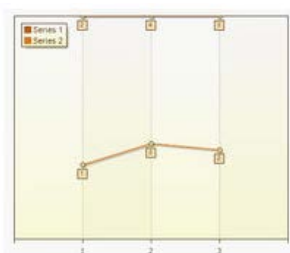
- **Пошаговые линии (Stepped Line).**



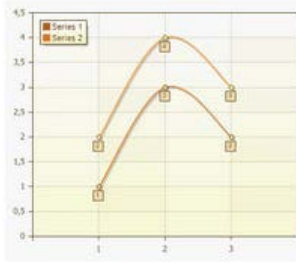
- **Линии с накоплением (Stacked Line).** Линии с накоплением, выводимые с маркерами, указывающими на отдельные значения данных, или без них, используются для демонстрации трендов вкладов каждой величины со временем или в зависимости от упорядоченных категорий. Линии без маркеров рекомендуется использовать при аппроксимации множества категорий значений.



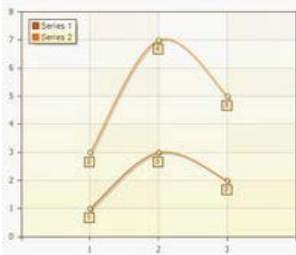
- **Нормированные линии с накоплением (Full-Stacked Line).** Нормированные линии с накоплением, выводимые с маркерами, указывающими на отдельные значения данных, или без них, используются для демонстрации вклада каждой величины в итоговое значение. Линии без маркеров рекомендуется использовать при аппроксимации множества категорий значений.



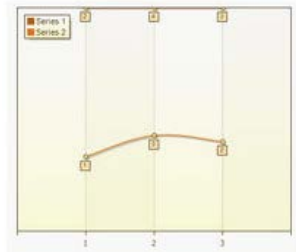
- *Плавные линии (Spline)*



- *Плавные линии с накоплением (Stacked Spline)*

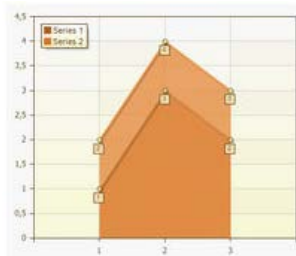


- *Нормированные плавные линии с накоплением (Full-Stacked Spline)*

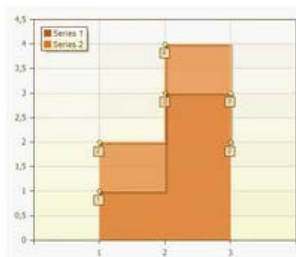


Диаграммы с областями (Area). Данные, которые расположены в столбцах или строках, можно изобразить в виде диаграммы с областями. Диаграммы с областями иллюстрируют величину изменений в зависимости от времени и могут использоваться для привлечения внимания к суммарному значению в соответствии с трендом. Например, данные, отражающие прибыль в зависимости от времени, можно отобразить в диаграмме с областями, чтобы обратить внимание на общую прибыль.

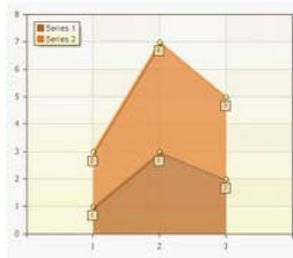
- *Область (Area).* Диаграммы с областями отображают тренды значений в зависимости от времени или категории.



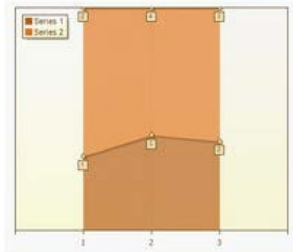
- *Пошаговая область (Stepped Area)*



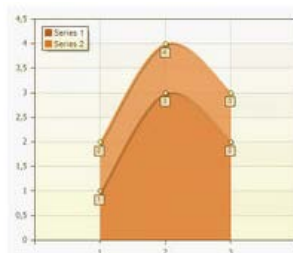
- *Область с накоплением (Stacked Area).* Области с накоплением отображают тренды вклада каждой величины в итог в зависимости от времени или по категориям.



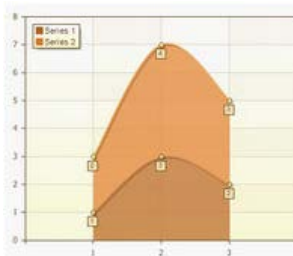
- **Нормированные области с накоплением (Full-Stacked Area).** Нормированные диаграммы с областями с накоплением отображают тренды вклада каждой величины, в процентах, в итог в зависимости от времени или по категориям.



- **Диаграммы со сглаженными областями (Spline Area).** Сглаженная область (Spline Area)



- **Сглаженная область с накоплением (Stacked Spline Area)**



- **Нормированные сглаженные области с накоплением (Full-Stacked Spline Area)**

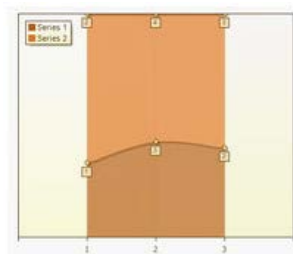
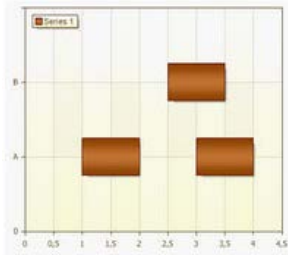


Диаграмма Ганта (Gantt Area). Диаграмма Ганта – это популярный тип столбчатых диаграмм, который используется для иллюстрации плана, графика работ по какому-либо проекту. Является одним из методов планирования проектов. *Диаграмма Ганта (Gantt).* Диаграмма Ганта представляет собой отрезки, размещенные на горизонтальной шкале времени. Каждый отрезок соответствует отдельной задаче или подзадаче. Задачи и подзадачи, составляющие план, размещаются по вертикали. Начало, конец и длина отрезка на шкале времени соответствуют началу, концу и длительности задачи.



Точечные диаграммы (Scatter Area). Данные, которые расположены в столбцах и строках, можно изобразить в виде точечной диаграммы. Точечная диаграмма отображает две группы чисел как один ряд координат x и y . Точечная диаграмма имеет две оси значений, при этом один набор значений выводится вдоль горизонтальной оси (оси X), а другой — вдоль вертикальной оси (оси Y). На точечной диаграмме эти значения объединяются в одну точку данных и выводятся с неравными интервалами, или кластерами. Точечные диаграммы обычно используются для представления и сравнения числовых значений, например, научных, статистических или инженерных данных.

Точечные диаграммы рекомендуется использовать, если:

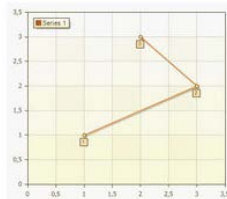
- Требуется изменять масштаб горизонтальной оси;
- Значения расположены на горизонтальной оси неравномерно;
- На горизонтальной оси имеется множество точек данных;
- Требуется продемонстрировать не различия между точками данных, а аналогии в больших наборах данных;
- Требуется сравнить множество точек данных без учета времени — чем больше данных будет использовано для построения точечной диаграммы, тем точнее будет сравнение.

Виды диаграмм:

- **Точечная диаграмма (Scatter).** Этот тип диаграммы позволяет сравнить пары значений. Точечные диаграммы с маркерами следует использовать, если данные определенным образом упорядочены.



- **Точечная диаграмма с линиями (Scatter Line).** Этот тип диаграммы может выводиться с прямыми линиями, соединяющими точки данных, или без них. Прямые линии могут выводиться с маркерами или без них.



- **Точечная диаграмма с плавными линиями (Scatter Spline).** Этот тип диаграммы может выводиться с плавными кривыми, соединяющими точки данных, или без них. Кривые могут выводиться с маркерами или без них. Если количество точек данных велико, используйте точечную диаграмму без маркеров.



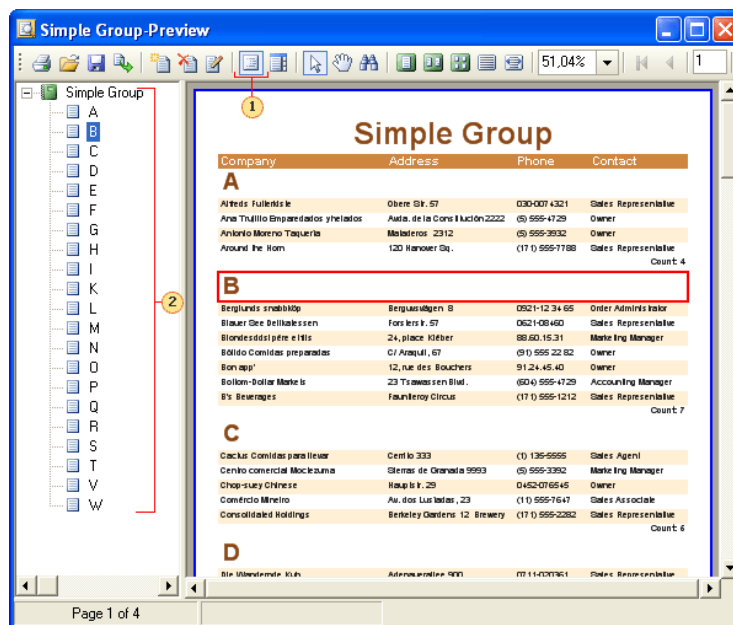
Навигация

Генератор отчетов предоставляет все необходимые инструменты для создания связей, как по отчету, так и из отчета на внешние объекты. Основными инструментами являются закладки и гиперссылки и дерево закладок в окне предварительного просмотра отчета. Для формирования закладок используется свойство *Закладка*. Для указания гиперссылок используется свойство *Гиперссылка*.

Оба этих свойства есть у всех компонентов как визуальных (текст, изображение, страницы и т.д.) так и у не визуальных (секции, контейнеры). Соответственно, между любыми компонентами отчета можно организовать гиперссылки и на любые компоненты можно делать закладки. Элементы управления, которые используются для построения диалоговых окон, не имеют указанных свойств и не могут формировать закладок или элементов навигации.

Закладки

Закладки предназначены для отображения структуры отчета. Также закладки используются для пометки компонента, чтобы потом можно было сделать на него ссылку при помощи гиперссылки. Все компоненты имеют свойство *Закладка*. Выражение, указанное в этом свойстве, в ходе построения отчета записывается в свойство *BookmarkValue*. Заполнение происходит в момент построения компонента. Это свойство невидимо в панели *Свойства*, но к нему можно обращаться из кода отчета или ссылаться на него из выражений. Перед отображением отчета в окне предварительного просмотра, генератор отчетов просматривает все компоненты готового отчета и заполняет дерево закладок.



Цифрой один на рисунке выше обозначено Дерево закладок в окне предварительного просмотра. Цифрой два - кнопка включения/выключения отображения дерева закладок. Примеры закладок:

- *Тест*;
- *МояКомпания*.

В свойстве допускается указывать и выражения, к примеру:

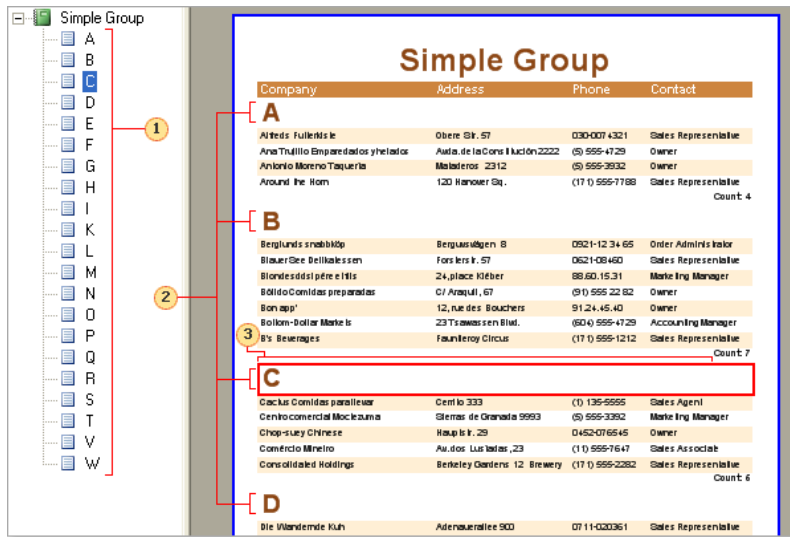
- *{Line}*;
- *{Товар.Наименование}*.

Дерево закладок

Дерево закладок позволяет отобразить иерархическую структуру отчета, позволяющую переходить к более подробному описанию при каждом переходе на нижестоящий уровень. Главным узлом в дереве закладок является узел, который отображает сам отчет. Наименование узла заполняется из свойства *Псевдоним отчета*. Поэтому, если есть необходимость

изменить наименование узла, нужно изменить это свойство. Если пользователь нажимает на один из узлов в дереве закладок, то генератор отчетов автоматически покажет компонент, к которому относится эта закладка. При этом компонент будет выделен красным цветом.

- ❖ **Важно!** В пределах одного уровня вложенности все закладки должны иметь уникальные наименования.



В левой части рисунка выше указано *Дерево закладок*, сформированное на основании значений свойств *BookmarkValue* каждого компонента. Значением свойства *BookmarkValue* является результат вычисления выражения из свойства *Закладка*. Цифрой два обозначены *Компоненты*, которые участвовали в формировании дерева закладок. После нажатия пользователем на узел в дереве закладок (цифра 3 на рисунке выше), генератор отчетов отображает соответствующий компонент. Компонент выделяется красной рамкой. Если закладка установлена компонент, который не отображается в окне предварительного просмотра, например секция, то она будет все равно подсвечена красным цветом.

Ссылки

Ссылки необходимы для организации навигации по отчету. Дополнительно для этого также могут использоваться свойства *Закладка* и *Тэг*. Гиперссылка указывается в свойстве *Гиперссылка*. Выражение, указанное в этом свойстве, в ходе построения отчета записывается в свойство *HyperlinkValue*. Заполнение происходит в момент построения компонента. Вариантов указания гиперссылок может быть три. Для разных случаев можно использовать разные варианты.

Ссылка на другой компонент в отчете с использованием закладок. В этом случае необходимо перед текстом гиперссылки поставить символ #. Это даст понять генератору отчетов, что эта ссылка внутри документа. Если в окне предварительного просмотра пользователь кликнет мышкой на этот компонент, то генератор отчетов начнет перебирать все закладки этого отчета и если совпадет закладка и гиперссылка (символ # в тексте гиперссылки отбрасывается), то в окне предварительного просмотра будет отображаться уже этот компонент. Необходимо помнить, что закладка отображается в дереве закладок. Свойство *Закладка* содержит текстовую метку, по которой этот компонент будет находиться во время обработки гиперссылки.

Ссылка на другой компонент в отчете с использованием тэга. В этом случае перед ссылкой добавляется два символа #. В этом случае поиск производится по Тэг свойствам компонентов (два символа # в тексте гиперссылки отбрасываются). Тэг свойства не отображаются в структуре отчета, если есть необходимость организовать навигацию без отображения закладок в структуре отчета или без изменения этой информации то используйте этот вариант. Ссылку на другой компонент в отчете с использованием свойства *Тэг* нельзя использовать при разработке отчетов для использования в *Asp.Net*.

Ссылка на внешние документы. В этом случае к гиперссылке не добавляется никаких специальных символов. Строка гиперссылки передается напрямую операционной системе для обработки. К примеру, для запуска приложения *Notepad* достаточно просто написать: *notepad.exe*. Для перехода по адресу в сети Интернет: *http://www.site.com*. Для ссылки на адрес электронной почты: *mailto: mail@domain.com*.

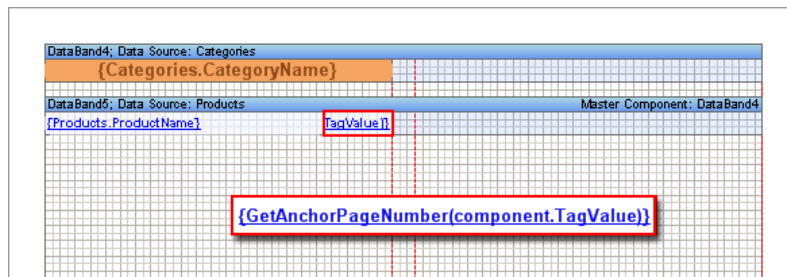
При построении *Web отчетов* закладки можно ставить только на видимые поля. Например, на текст, изображение. Иначе ссылка будет проигнорирована. Это же правило надо учитывать при экспорте отчетов в другие форматы.

Отчеты с оглавлением

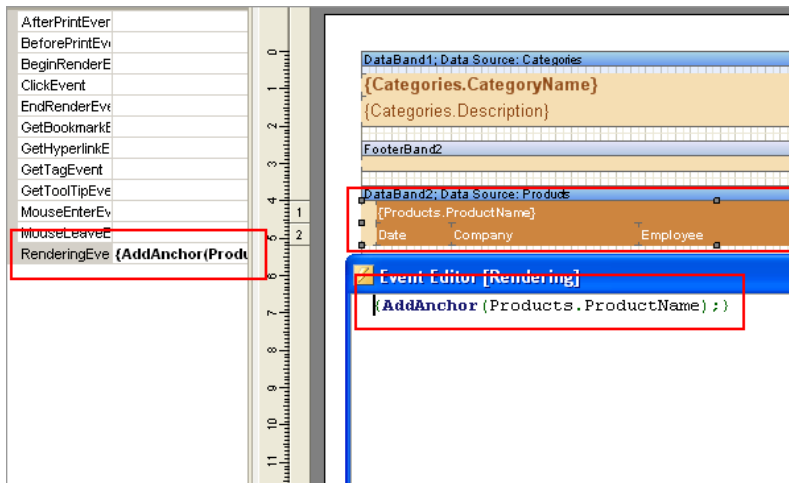
Часто возникает необходимость построить отчет с оглавлением. В таком отчете сначала необходимо вывести структуру отчета, а затем сам отчет. В этом случае встает вопрос, как вывести номера страниц, ведь в тот момент, когда происходит построение оглавления, еще не известны номера страниц, на которые ссылаются элементы оглавления. Для решения этой проблемы предназначен специальный элемент - *якорь*.

Для создания якоря применяется метод *AddAnchor*. В момент создания якоря генератор отчетов запоминает текущую страницу и сопоставляет ее с якорем с указанным названием. К примеру: *AddAnchor("MyAnchorName")* - в этом коде будет создан новый якорь с названием *"MyAnchorName"*. Для того чтобы получить значение якоря применяется метод *GetAnchorPageNumber*. Метод возвращает номер страницы по названию якоря. Если якоря с таким названием нет, то возвращается *0*. К примеру: *{GetAnchorPageNumber("MyAnchorName")}* - это текстовое выражение вернет номер страницы, по якорю с названием *"MyAnchorName"*. Таким образом, имея названия якоря можно узнать номер страницы, на которой он был создан. При помощи этих двух методов и организуется построение оглавления. Для этого сначала строится оглавление, вместо номеров страниц вставляются ссылки на якоря. Причем для всех компонентов, которые вызывают функцию для получения номера страницы через якорь, необходимо установить свойство *Обработать* в конце в *Да*. Это необходимо, для того чтобы эти компоненты были обработаны в самом конце отчета, когда все номер страниц уже известны.

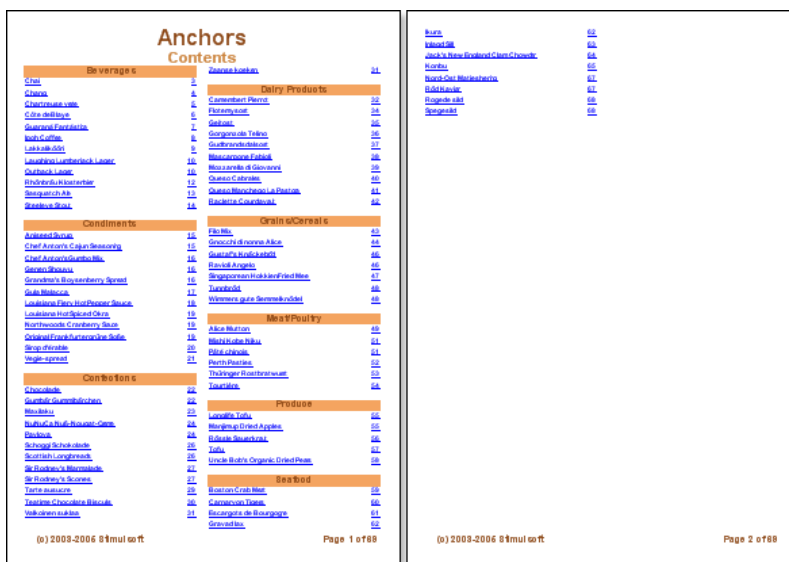
После построения оглавления строится сам отчет. В ходе построения создаются якоря. После построения отчета вместо ссылок на якоря в содержании автоматически расставляются реальные номера страниц. У текстового компонента, выражение которого возвращает номер страницы, необходимо включить свойство *Обработать* в конце. Это необходимо, для того чтобы значения этих текстовых компонентов обработались после построения отчета (когда уже будут известны номера страниц). В свойстве *Text* указывается следующее текстовое выражение: *{GetAnchorPageNumber(component.TagValue)}* - это текстовое выражение вернет номер страницы при помощи якоря.



В качестве наименования якоря используется значение свойства *Тэг*. Для заполнения свойства *Тэг* используется выражение: *{Products.ProductName}* - в данном случае используется наименование продукта. Поэтому нельзя использовать, к примеру, такое выражение: *{GetAnchorPageNumber(Products.ProductName)}*, т.к. компонент, который содержит выражение, будет обрабатываться в конце построения отчета, то значение поля *Products.ProductName* будет для всех строк одинаковым – последним в списке. Поэтому необходимо запомнить значения поля *Products.ProductName* для каждой строки в тот момент, когда будет строиться оглавление. Для этого и используется свойство *Тэг*. На второй странице строится сам отчет. В событии *Построение у компонента* секция *Данные*, по которой строится оглавление, вызывается метода *AddAnchor*. Этот метод запомнит текущую страницу в момент своего вызова.

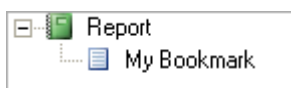


Наименованием якоря является значение поля *Products.ProductName*. В итоге сначала строится первая страница. Затем строится вторая страница, запоминаются номера страниц. После построения отчета, движок генератора отчетов возвращается к первой странице и расставляет номера страниц.

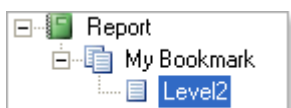


Формирование закладок при помощи кода

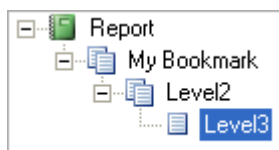
При помощи свойства *Закладка* можно сформировать достаточно сложную структуру закладок в отчете. Однако часто возможностей предоставляемых этим свойством оказывается недостаточно. К примеру, может понадобиться добавить в дерево закладок свои узлы без использования свойства *Закладка*. Или закладку необходимо расположить в другом уровне вложенности. Для этих целей генератор отчетов предоставляет свойство *ManualBookmark*. Это невидимое свойство отчета и оно доступно только из кода. Использовать это свойство достаточно просто. К примеру, для того чтобы добавить закладку первого уровня вложенности можно использовать такой код: *ManualBookmark.Add("My Bookmark")*. Этот код создаст такую закладку в дереве закладок:



Для того чтобы добавить в дерево закладку второго уровня необходимо написать такой код: *ManualBookmark["My Bookmark"].Add("Bookmark Level2")*.



Соответственно для третьего уровня: `ManualBookmark["My Bookmark"]["Level2"].Add("Bookmark Level3")`.



Чтобы создать все три закладки сразу можно воспользоваться только последним приведенным примером кода. Генератор отчетов автоматически проверит наличие каждой из закладок в дереве и добавит недостающие. При использовании свойства *Manual Build*, закладки создаваемые при помощи свойства *Закладка* не отображаются в дерево закладок. При помощи закладок может понадобиться организовать навигацию. Для того чтобы генератор отчетов мог найти соответствующие компоненты, необходимо заполнить свойство *Закладка* этих компонентов. При этом значение свойства *Закладка* должно совпадать с наименованием создаваемой закладки. Например, добавляем закладку: `ManualBookmark.Add(Customers.CompanyName)`. Соответственно у компонента свойство *Закладка* заполняем так: `{Customers.CompanyName}`. В результате все компоненты будут помечены закладкой с названием компании. В дерево будут добавлены эти же названия компаний. И при нажатии на узлы дерева закладок компоненты будут успешно находиться.

Формирование отчетов по изменению значений

Представим архивные значения переменной в виде таблицы из двух столбцов, первый из которых содержит время прихода сигнала и фиксации значения, второй – соответствующее этому моменту времени значение переменной. При формировании отчета архивных значений одной переменной по изменению в данную таблицу будет дописываться новая строка каждый раз, когда будет поступать новое значение. При формировании отчета по архивным данным двух или более переменных возникают некоторые сложности. Данные о значениях могут поступать в различные моменты времени (асинхронно), поэтому в таких случаях требуется задать некоторые условия, с помощью которых генератор отчетов сформирует итоговую таблицу, содержащую значения всех переменных.

Одним из таких условий является задание величины мертвой зоны. Если несколько переменных были сохранены в архиве приблизительно в одно время, то для представления пользователю желательно считать эти измерения одновременными. С этой целью времена записи, интервал времени между которыми не превышает величины мертвой зоны, считаются одним моментом времени, а сами эти значения будут записаны в одну строку таблицы. Для задания величины мертвой зоны необходимо в дереве словаря выделить *Источник данных* и перейти к его свойствам. Далее ввести время в секундах в поле *Мертвая зона*. Чтобы создать отчет по изменению для архивных значений двух или более переменных, необходимо добавить источник данных – *Архив MasterSCADA*. Далее нужно в дереве словаря выделить *Источник данных* и перейти к его свойствам. Необходимо для свойства *Способ формирования значений* выбрать *По изменению*. Формирование таблицы значений двух или более переменных будет при этом происходить следующим образом: каждый раз для построения новой строки таблицы из всех архивных значений, ещё незаписанных в таблицу, всех переменных генератор отчета будет выбирать момент времени максимально приближенный к началу формирования отчета (попросту самый ранний) и запишет его в первый столбец новой строки.

Может оказаться, что данному моменту времени соответствуют значения не всех переменных, и тогда в некоторых столбцах таблицы генератору отчетов будет неясно, какое же из имеющихся значений нужно записать. Для разрешения подобной ситуации необходимо задать один из двух вариантов для свойства *Обработка* каждой из колонок: *Последнее значение* и *Точное значение*. Если задан вариант *Последнее значение*, то будет выбрано последнее полученное (зафиксированное) к данному моменту времени значение переменной. Если оно отсутствует, то будет записан ноль, или клетка останется пустой. Если же для свойства *Обработка* будет выбран вариант *Точное значение*, то в строку, соответствующую выбранному моменту времени значение из этой колонки будет записано только в том случае, если оно в точности (до величины мертвой зоны) получено в этот момент времени. Иначе будет записан ноль или клетка таблицы останется пустой. Количество архивных значений для разных переменных может отличаться, у одной их может быть больше (значения фиксировались чаще), у другой меньше (фиксировались реже). В таком случае можно сделать так, что-

бы метки времени значений одной из колонок игнорировались при выборе времени следующей строки. Для этого необходимо перейти к свойствам колонки и для свойства *Основной* выбрать *False*. Если у свойства *Основной* выбран *False*, то данная колонка не участвует в определении времени следующей строки, и её значения будут записаны в таблицу в зависимости от настроек обработки.

Свойства переменных источника данных

В свойствах конкретной переменной источника можно задать ее поведение, атрибут получения, способ обработки.

Основной - присутствует если способ формирования установлен *По изменению*. Если данная настройка включена, то каждое изменение этой переменной приводит к созданию строки данных в отчете.

Атрибут данных - определяет какой из атрибутов переменной выводится в отчет. Каждая переменная источника имеет три атрибута:

- *Значение* – значение переменной в архиве;
- *Время* – метка времени переменной в архиве;
- *Признак качества* – признак качества переменной в архиве.

Обработка - определяет способ обработки параметра. Каждый из способ описан в разделе Формирование отчета по периодической выборке значений.

Граница интервала - определяет, необходимо ли учитывать граничные значения интервала.

Модуль графиков и журналов сообщений

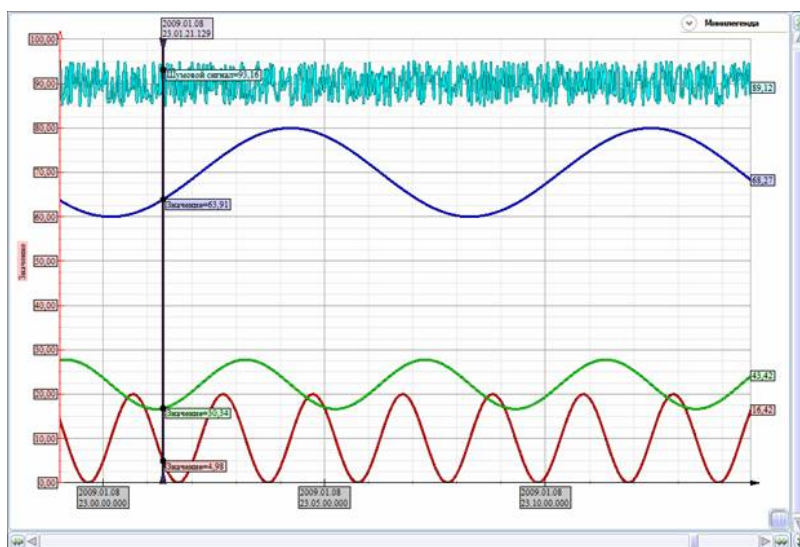
Данный раздел делится на три части: первая часть – методические указания по работе с трендом, описывает сценарии и типовые операции работы с трендом. Вторая часть – методические указания по работе с журналом, описывает сценарии и типовые операции работы с журналом. Третья часть – общее справочное описание, является документацией разработчика и детально описывает все настройки и элементы пользовательского интерфейса.

Методические части построены в разрезе работы оператора. Справочная часть построена в разрезе работы разработчика проекта. Для разъяснения непонятных терминов методических частей следует обращаться к содержанию справочной части.

Методические указания по работе с трендом

Общие возможности

Для визуализации графиков зависимости параметров от времени, используется документ типа *тренд*, который позволяет не только просматривать эти графики в различных масштабах, но и анализировать их. На рисунке ниже приведен пример тренда с отображением четырех параметров.



В дереве объектов проекта создание, просмотр и редактирование документов и окон трендов каждого объекта производится с закладки *Тренды*. Наряду с этим можно включать тренды в состав мнемосхем и других графических документов и окон. Важной особенностью модуля тренда является возможность включения в тот же документ таблицы значений, журнала сообщений и синхронизации просмотра значений, сообщений и трендов.

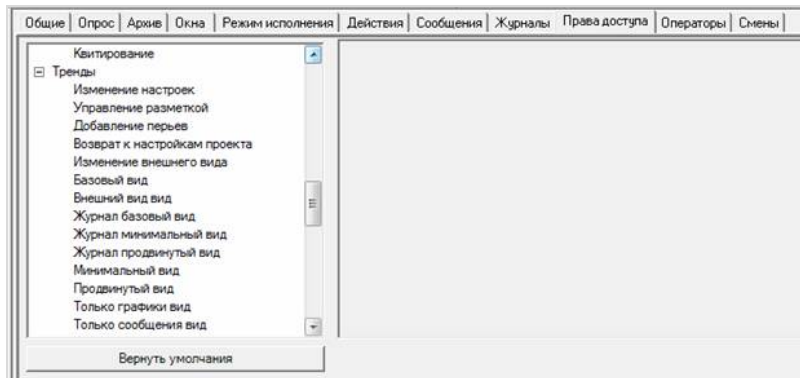
Весь описываемый пользовательский интерфейс тренда является полностью настраиваемым, в том числе при наличии соответствующих прав доступа в режиме исполнения, что позволяет найти тот компромисс между простотой работы и уровнем функциональности, который необходим для комфортной работы оператора в рамках конкретного проекта. Переключение между уровнями сложности пользовательского интерфейса производится из контекстного меню графической области, таблицы значений или таблицы сообщений.

Уровни сложности определяют наличие визуальных элементов (интерфейс пользователя). Отличия предопределенных уровней сложности следующие:

- *Продвинутый уровень* является наиболее полным уровнем с возможностью настройки большого количества свойств, предназначен для продвинутых пользователей и детальной настройки. Содержит область графиков, таблицу значений, таблицу сообщений, легенду и панель настроек;
- *Базовый уровень* содержит основные визуальные элементы, предназначен для использования без предварительного ознакомления с документацией. Содержит область графиков, таблицу значений, легенду и панель настроек;

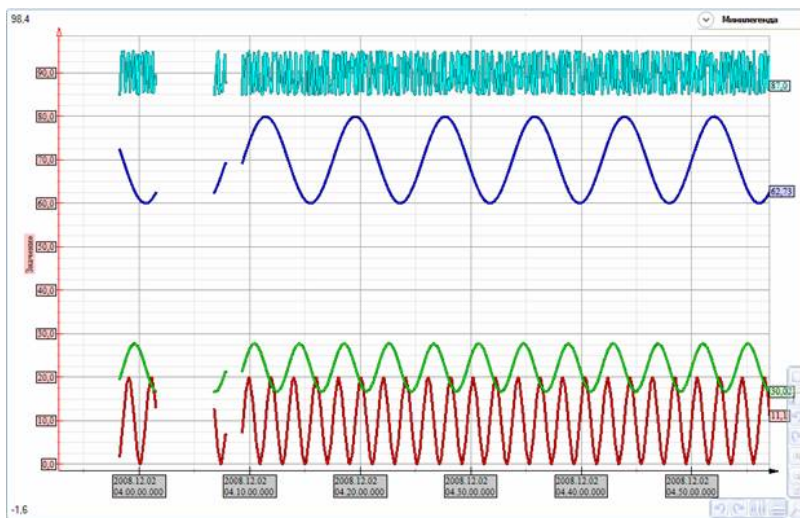
- *Минимальный уровень* содержит минимально необходимый набор визуальных элементов для просмотра графиков, в нем полностью отсутствует возможность настройки, реализован минимум функциональности для неподготовленных пользователей. Содержит только область графиков;
- *Уровень Только графики* содержит только область для графического представления перьев. Содержит только область графиков;
- *Уровень Таблица значений* позволяет отображать данные только в табличном виде.

Возможно создание пользовательских уровней сложности на основе существующих с помощью специального *редактора уровней сложности*. Для переключения на выбранный уровень - текущий оператор должен иметь соответствующие права. При отсутствии прав - уровень недоступен. Права оператора на переключение уровней сложности задается на уровне компьютера, на закладке *Права доступа*.



Окно тренда состоит из нескольких функциональных панелей:

- *Окно графиков*. Служит для отображения графических объектов – перьев, элементов разметки, средств анализа;



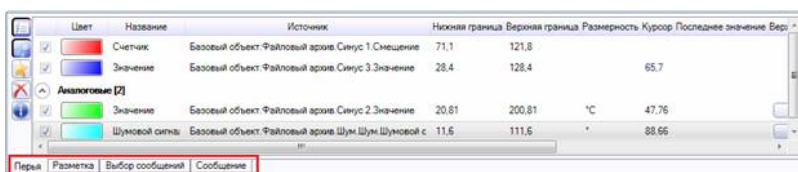
- *Таблица значений*. Служит для отображения данных в табличном виде;

Режим интерполяции		
Дата/время	Значение %	Значение °C
06.01.2009 15:35:18.128	18,91	48,53
08.01.2009 22:34:23.863	18,66	48,66
08.01.2009 22:34:24.799	18,39	48,79
08.01.2009 22:34:25.813	18,09	48,91
08.01.2009 22:34:26.827	17,77	49,03
08.01.2009 22:34:27.841	17,43	49,14
08.01.2009 22:34:28.855	17,07	49,24
08.01.2009 22:34:29.869	16,69	49,34
08.01.2009 22:34:30.883	16,29	49,43
08.01.2009 22:34:31.912	15,88	49,51
08.01.2009 22:34:32.926	15,45	49,59
08.01.2009 22:34:33.956	15,00	49,66

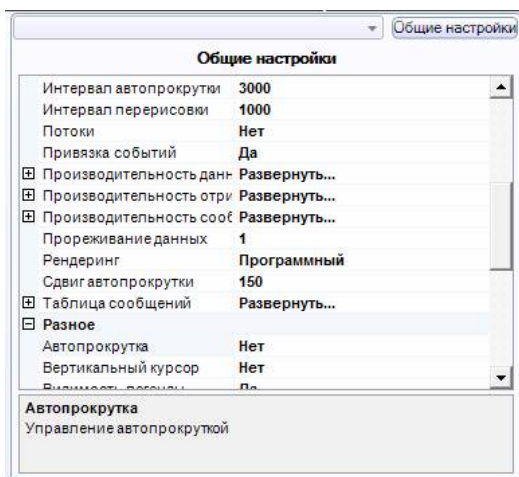
- *Таблица сообщений*. Служит для отображения сообщений в табличном виде;



- **Панель легенды.** Служит для хранения и управления списками объектов тренда. Легенда содержит закладки, ведущие на дополнительные панели: перья (*панель перьев*), разметка (*панель разметки*), выбор сообщений (*панель выбора сообщений*), сообщение (*панель информации о сообщении*);



- **Панель настроек.** Служит для изменения настроек объектов тренда.



Во избежание загромождения пользовательского интерфейса предусмотрена возможность управления видимостью панелей (разрешить или запретить видимость панели на экране). **Управление видимостью панелей** тренда производится из *тулбара тренда* при помощи кнопок с фиксацией.



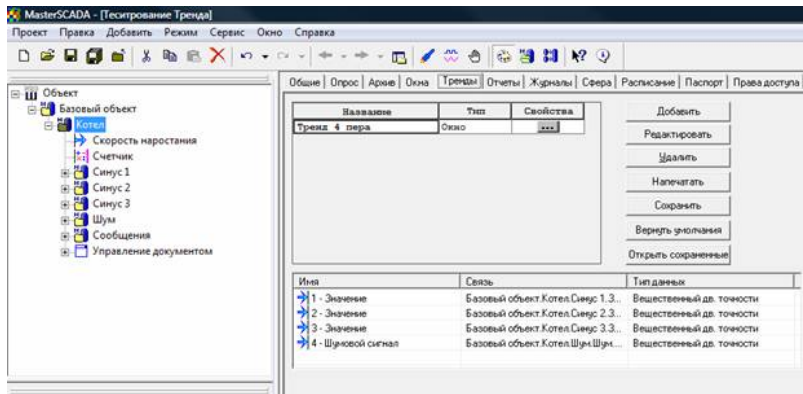
- Кнопка *Графики* - управление видимостью окна графиков;
- Кнопка *Значения* - управление видимостью таблицы значений;
- Кнопка *Сообщения* - управление видимостью таблицы сообщений;
- Кнопка *Легенда* - управление видимостью панели легенды;
- Кнопка *Настройки* - управление видимостью панели настроек.

Видимость этих кнопок, как и самих панелей может меняться в зависимости от уровня сложности.

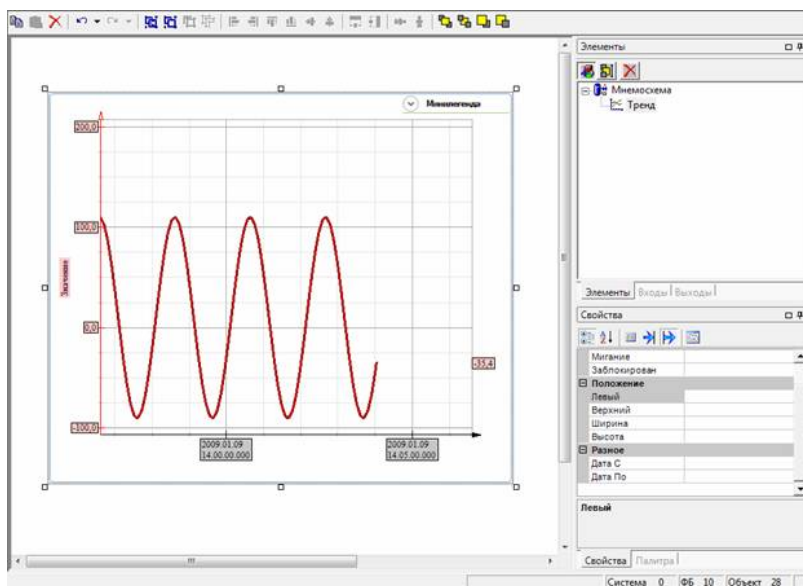
Тренд можно добавить в проект двумя способами:

- Добавление документа *Тренд* с закладки *Тренды* на уровне объекта *дерева объектов*;
- Добавление элемента *Тренд* на редактируемую мнемосхему как *ActiveX* элемент из палитры мнемосхемы.

В дереве объектов проекта тренды каждого объекта доступны с закладки *Тренды*. На рисунке иже показан доступ к трендам объекта *Котел*. Справа показаны кнопки управления списком трендов объекта.

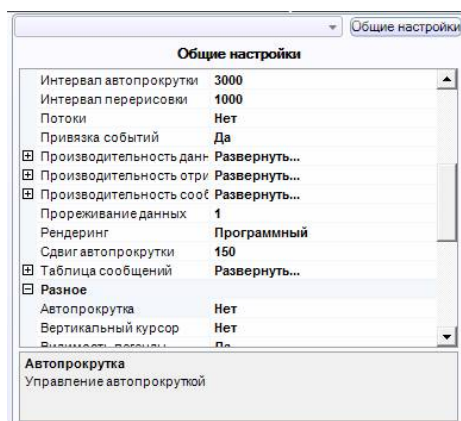


Тренд может быть добавлен на мнемосхему как любой ActiveX элемент. При добавлении на мнемосхему для тренда по умолчанию устанавливается максимально упрощенный *уровень сложности* пользовательского интерфейса *Только графики*, однако он может быть изменен в процессе настройки.



При добавлении тренда на мнемосхему становится доступной динамизация текущих границ времени тренда - настроек *Дата С* и *Дата По*.

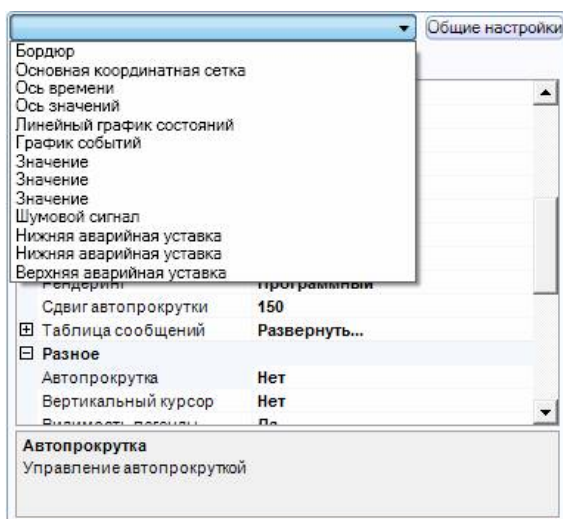
Настройка содержимого тренда может производиться только в *продвинутом уровне сложности*, в *базовом уровне сложности* или *пользовательских уровнях*, унаследованных от них. В режиме настроек видна *панель настроек*, которая служит для отображения настроек текущего выбранного объекта.



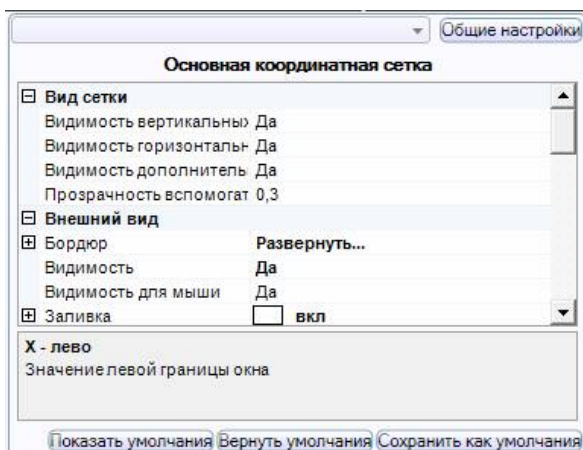
Выбор графического объекта для изменения его настроек может производиться следующими способами:

- Кликом мыши в *окне графиков* по объекту;
- Выбором объекта в *панели перьев* или *панели разметки*;

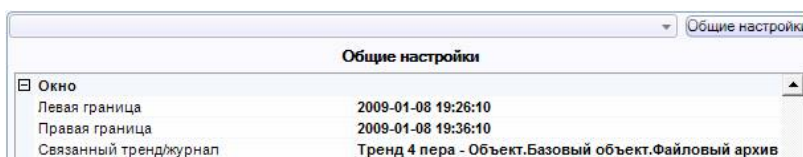
- Выбором объекта из выпадающего списка в панели настроек (только для режима тренда).



Есть возможность выбора нескольких объектов при зажатой кнопке *Ctrl*. В панели настроек существует возможность работать с predetermined настройками (умолчаниями) – они заданы для всех графических объектов. Редактирование умолчаний позволяет установить значение настройки одновременно для всех объектов данного типа, если это значение не было переопределено в объекте. Настройки, которые были переопределены, выделяются жирным шрифтом.



Два тренда (и больше, по цепочке) можно *связать друг с другом по временной шкале*. При изменении горизонтальных границ окна одного тренда – будут одновременно меняться границы второго тренда. Это может быть удобно, когда хочется просматривать разные перья в одинаковом масштабе времени, но на отдельных трендах. Связь можно установить только для уже открытых трендов. Необходимо сначала открыть два тренда, а потом в одном из них установить свойство *Общие - Связанный тренд*. После установки связи – она сохранится до ее разрыва (обнуления данного свойства).



Работа с перьями

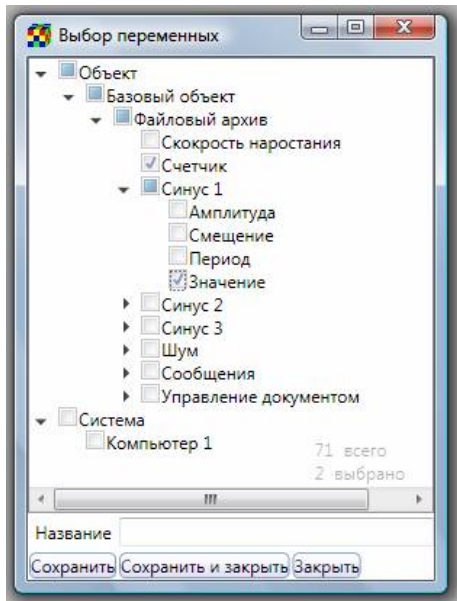
Перо является базовым понятием MasterSCADA. *Перо* – это график, с которым связана переменная MasterSCADA из дерева объектов.


Чтобы добавить на тренд график еще одного параметра, есть два варианта:

- Перетаскивание (*Drag&Drop*) элемента дерева объектов MasterSCADA в *Окно графиков*, в *Легенду* или в *Минилегенду*. При перетаскивании элемента – будут добав-


лены все его подэлементы. При перетаскивании переменной на уже имеющееся перо в *Легенде* или *Минилегенде* – будет произведена перепривязка источника данных для этого пера;

- Добавление перьев из диалога *Выбора перьев*.

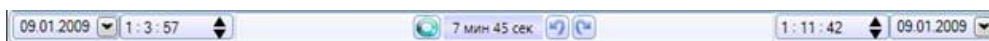


Данный диалог можно вызвать нажатием на кнопку  выбора перьев. Эта кнопка расположена в *тулбаре тренда*, а также в *тулбаре панели перьев* у легенды. Также этот диалог можно вызывать из контекстного меню окна *графиков*.

Удаление перьев производится следующими способами:

- В *диалоге выбора перьев*;
- В *тулбаре панели перьев* кнопкой ;
- Из контекстного меню тренда в режиме настроек командой *Удалить выделенные объекты*.

Активное перо является ключевым понятием тренда. На тренде одновременно может находиться несколько перьев и у каждого может быть своя вертикальная шкала и вертикальный масштаб отображения. Горизонтальные оси всех графиков совмещены и их границы совпадают с интервалом на *панели даты-времени*.




В каждый момент времени может быть активным только одно перо – именно его шкала отображается на вертикальной оси по умолчанию. Активным перо можно сделать следующими способами:


- Выбрать перо в *Окне графиков* кликом мыши по геометрии графика;
- Выбрать перо в *Панели перьев* кликом мыши по названию пера;
- Выбрать перо в *Минилегенде* кликом мыши по названию пера.

Активное перо подсвечивается утолщением геометрии графика, а также подсветкой в панели перьев и минилегенде. По умолчанию вертикальная ось принимает цвет активного пера, а разметка координатной сетки подгоняется так, чтобы значения активного пера были округленными. Активное перо является текущим выделенным графическим объектом и его свойства отображаются в панели настроек. Активное перо и выделенный объект – это не одно и то же. Выделение объектов производится для выполнения действий с ними, а установка активного пера переключает масштаб вертикальной оси значений.

Масштабирование перьев – это операция сжатия/растягивания по вертикали графиков данных.

Следующие способы масштабирования применяются к выделенному графику (если ни один график не выделен – то масштабируются все):


- При помощи кнопки подбора автоматического масштаба пера  на *панели Вертикального масштаба* или в панели перьев;



- При помощи кнопки выбора масштаба шкалы  на панели вертикального масштаба или в панели перьев;
- При помощи мыши: прокрутка колесом при зажатой правой кнопке масштабирует перо под мышью. Если под мышью нет пера, то масштабируются все перья.

Следующие способы применяются к активному графику:



- При помощи *Элементов управления вертикальными границами*;
- При помощи установки текущих вертикальных границ в панели перьев.

Следующие способы применяются ко всем графикам:

- При помощи кнопки *Лула* ;
- При помощи *вертикального навигатора* (визуальный элемент, аналогичный скроллбару, но с расширенной функциональностью).

Отмена изменений вертикального масштаба перьев осуществляется при помощи кнопок отмены изменения вертикальной оси   на *панели вертикального масштаба*.

Вертикальное **размещение перьев** относительно друг друга доступно в продвинутом уровне сложности в режиме настроек. Изменение вертикального расположения перьев в окне графиков возможно следующим образом:

- При помощи мыши. Следует при зажатой кнопке переместить мышь по вертикали в окне графиков, затем отпустить кнопку. Если под мышью в момент перемещения находился график – то будет перемещен только он, иначе все графики будут сдвинуты относительно изначального положения;
- При помощи кнопок *Сдвинуть вверх* и *Сдвинуть вниз*   на панели вертикального масштаба.

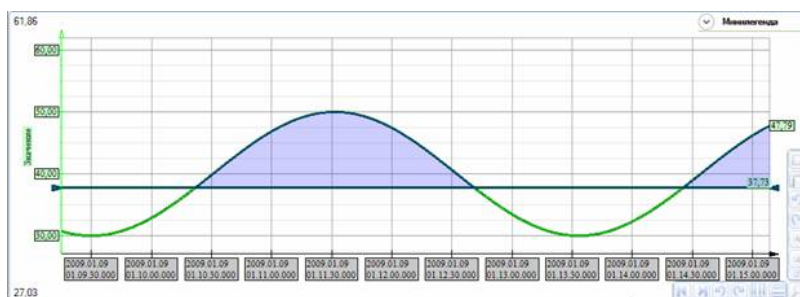
Настройка перьев производится в *режиме настроек*. Для пера много настроить большое количество характеристик, наиболее важными из которых являются:

- Произвольное форматирование выводимых значений пера и оси X. Задается в свойствах;
- Вид связанной с пером координатной сетки: произвольная разметка, выравнивание округление;
- Внешний вид пера, цвет линий, заливки, графических точек.

Также для пера и связанной с ним оси может быть задана логарифмическая шкала настройкой *Шкала Y*.

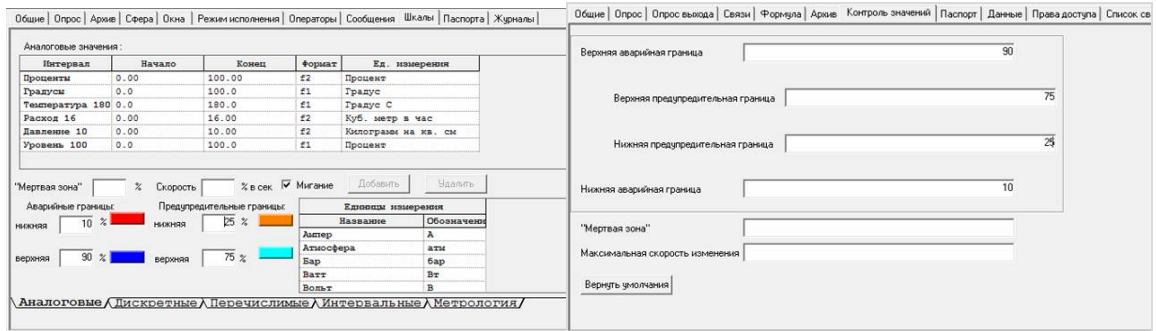
Работа с уставками перьев

Уставки – это линии контрольных границ, способные отображать переход значения пера за границу.



Границы пера могут задаваться двумя способами:

- Из системной шкалы, привязанной к параметру (шкала конфигурируется на уровне узла *Система* дерева системы MasterSCADA);
- На вкладке *Контроль значений* параметра MasterSCADA.



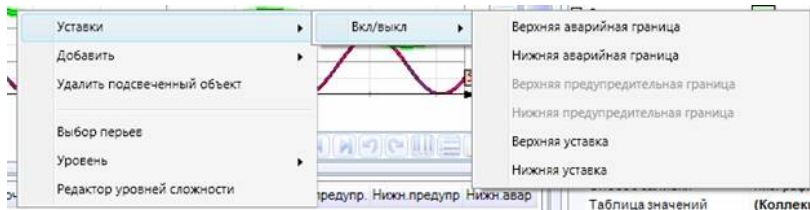
Существует четыре границы:

- Верхняя аварийная;
- Верхняя предупредительная;
- Нижняя предупредительная;
- Нижняя аварийная.

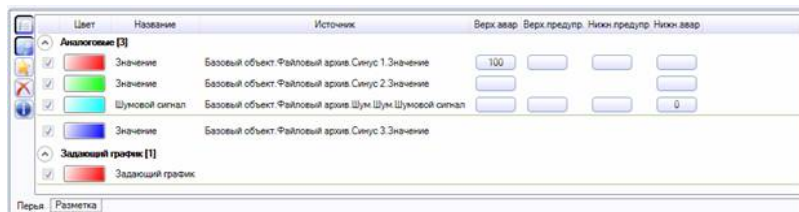
Во избежание загромождения окна графиков – уставки отображаются только у *активного пера*. Графически уставки могут быть представлены как: верхняя/нижняя, что определяет направление заливки при переходе значения пера через заданную границу. Область действия уставки (заливка перехода значение через уставку) может быть до следующей уставки или до края экрана со смешиванием цветов. Тип заливки уставки может быть внутренний или внешний – в первом случае заливается площадь между линией уставки и геометрией пера, во втором между краем экрана и геометрией пера.

Управление отображением уставки пера может производиться следующими способами:

- Из *контекстного меню окна графиков* для активного графика. Уставки, границы которых не определены у пера, подсвечены серым цветом и их добавить нельзя;



- Из *панели перьев* кнопками включения уставки.

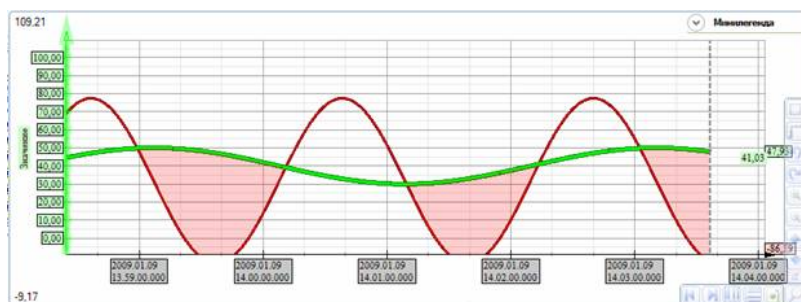


Эти кнопки присутствуют только для заданных уставки. Включенные на отображение уставки показаны значением уставки на кнопке. В *режиме настроек* возможно добавление неограниченного количества пользовательских уставки командами контекстного меню окна графиков *Верхняя уставка* и *Нижняя уставка*. После добавления пользовательской уставки необходимо произвести ее настройку. Пользовательские уставки в отличие от уставки шкалы или уставки контроля значений используются только на тренде и не порождают сообщений о нарушении.


На закладке *Контроль значений* у параметра MasterSCADA можно задавать параметрические значения уставки перетаскиванием другого параметра в элемент ввода границы.

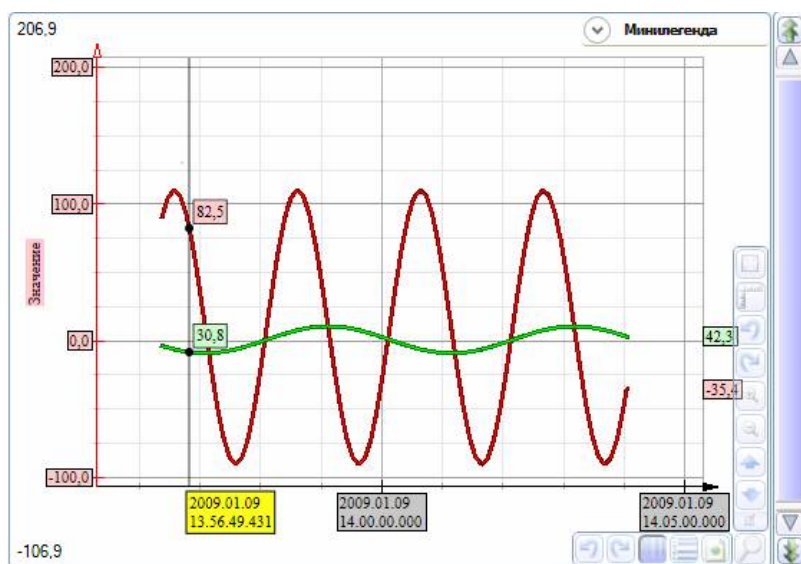
Верхняя аварийная граница	=Объект.Базовый объект.Файловый архив.Синус 1.Значение
Верхняя предупредительная граница	75.000000
Нижняя предупредительная граница	25.000000
Нижняя аварийная граница	10.000000
"Мертвая зона"	
Максимальная скорость изменения	
<input type="button" value="Вернуть умолчания"/>	



Параметрические уставки также способны отображаться на тренде.



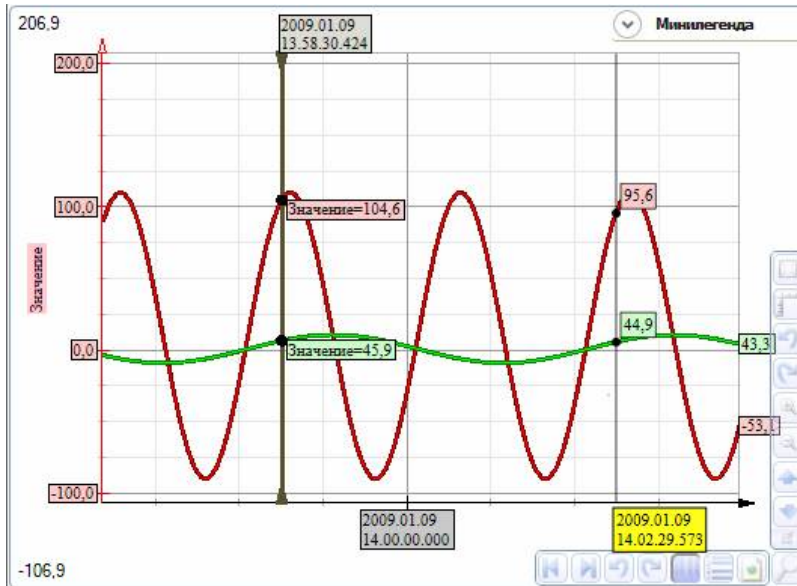
Графические курсоры предназначены для получения срезов графиков в заданный момент времени или по заданному значению. Также графические курсоры предоставляют дополнительный пользовательский интерфейс по установке закладок и заданию пользовательских уставок.


Вертикальный курсор отображает значения (временные срезы) графиков и других объектов, с которыми он пересекается. Режим вертикального курсора переключается с панели горизонтального масштаба кнопкой управления вертикальным курсором . Положение курсора по X отображается внизу линии курсора.

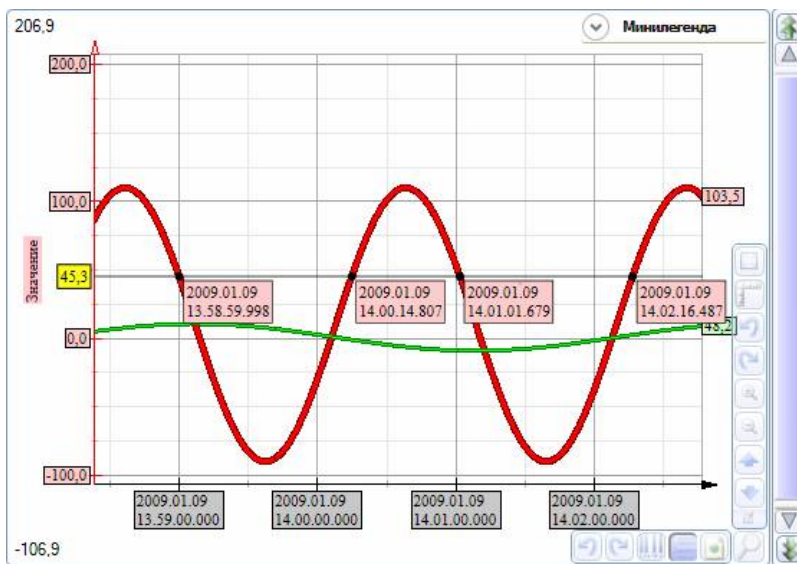


Двойной клик мыши в режиме вертикального курсора устанавливает закладку. **Закладка** - это вертикальная линия, привязанная ко времени. Закладка показывает временные срезы графиков. Также закладки служат для облегчения навигации при просмотре архивной истории. Навигация между закладками осуществляется при помощи кнопок перехода к предыдущей и следующей закладке   в панели горизонтального масштаба.

Также закладка может быть добавлена из контекстного меню тренда.



Горизонтальный курсор отображает время (срезы значений) графиков и других объектов, с которыми он пересекается. Режим горизонтального курсора переключается с панели горизонтального масштаба кнопкой управления горизонтальным курсором . Положение курсора в шкале активного графика по Y отображается слева линии курсора.



Двойной клик мыши в режиме горизонтального курсора добавляет пользовательскую уставку для активного графика. Следует учесть, что добавление пользовательских уставок доступно только в режиме настроек.

Работа с интервалами отображения графиков

Листание графика – это операция сдвига границ времени графика без изменения масштаба времени. Листание графика может осуществляться следующими способами:

- При помощи мыши. Следует в окне графиков нажать левую кнопку мыши, не отпуская сдвинуть мышь, затем отпустить кнопку;
- При помощи слайдера горизонтального навигатора, который при использовании в данном назначении аналогичен простому скроллбару.



Изменение масштаба времени происходит одновременно для всех элементов тренда, оперирующих понятием *время* (в том числе и для графиков). Управление масштабом времени может происходить тремя способами описанными ниже.

При помощи горизонтального навигатора. Навигатор включает в себя кнопки перехода к самому первому/последнему событию/значению за всю историю, кнопки сдвига временного

окна к более ранним/поздним данным на 10%, кнопки сдвига временного окна к более ранним/поздним данным на 100%, кнопки изменения левой/правой границ окна, ползунок сдвига временного окна

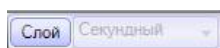
Посредством панели даты/времени. Панель даты/времени содержит следующие элементы: элемент *календарь* для выбора даты левой/правой границы, элемент *время* для выбора времени левой/правой границы, кнопка *управления автопрокруткой*, элемент *быстрого выбора интервалов времени*, кнопку *загрузки сохраненного состояния тренда* (состояния, сохраненного в режиме разработки), кнопку *возврата временного состояния RT* после загрузки сохраненного состояния тренда.



Рядом с элементами выбора левой и правой границ – находятся чекбоксы фиксации значений границ. При установке фиксации – граница при любых действиях остается неизменной, чтобы бывает полезно в сценарии *сжатия/растяжения* графиков.

Прокруткой колеса мыши при нахождении курсора в окне графиков осуществляется изменение масштаба времени окна графиков

Также при изменении масштаба времени большую роль играет панель выбора *слоев данных*.

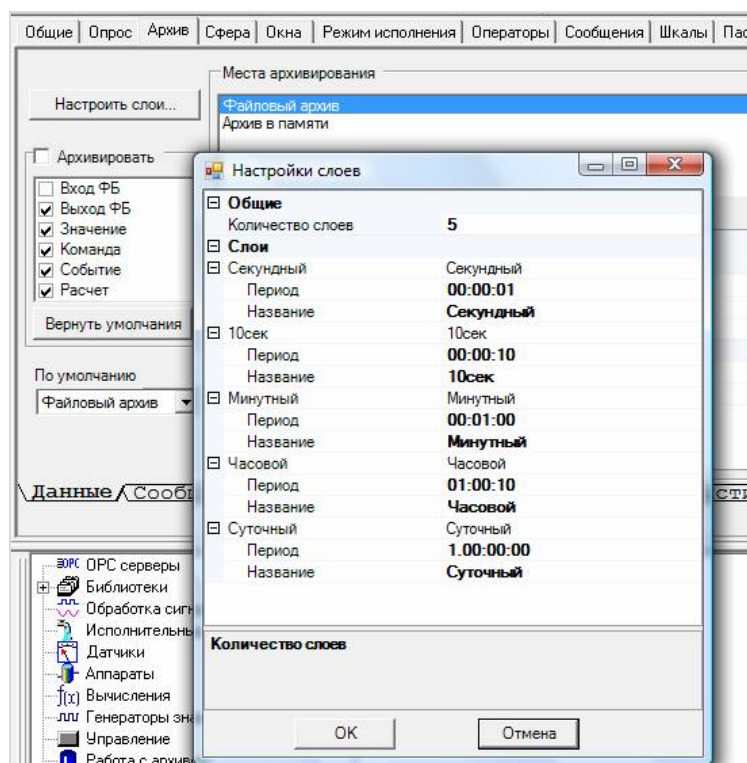



Слои данных – это заданные интервалы прореживания данных. За счет использования слоев при просмотре истории на больших интервалах времени из архива выбирается значительно меньший (на порядки) объем данных, что снижает нагрузку процессора и ускоряет работу тренда при просмотре архивной истории. Например, при просмотре архива данных за месяц – из архива выбираются не все данные, а прореженные данные за час (четыре значения за сутки: первое, максимальное, минимальное, последнее).

Слои данных выбираются автоматически, однако есть возможность просмотреть данные в определенном слое при помощи *панели выбора слоев данных*:

- *Непрореженный слой* отображает непрореженные данные;
- *Временный слой* – отображает данные, еще не помещенные в архив (например, при использовании отдельного архивного сервера, если с ним пропала связь).

Остальные слои могут настраиваться пользователем. Настройка производится на уровне узла *Система* дерева системы MasterSCADA с закладки *Архив* кнопкой *Настроить слои*.



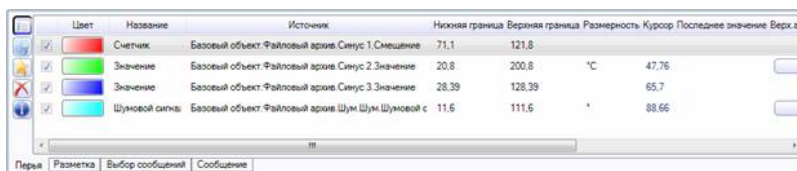
Режим автопрокрутки предназначен чтобы в режиме исполнения отображать на экране всегда самые актуальные данные. При включенном режиме автопрокрутки текущее время и соответственно данные всегда находятся в области видимости. В режиме автопрокрутки происходит отключение некоторых визуальных элементов, например, элементов задания даты/времени и кнопок навигации по маркерам. Режим автопрокрутки переключается кнопкой автопрокрутки  на *панели даты-времени*. Эта кнопка видна только в режиме исполнения.

Режим воспроизведения истории предназначен для воспроизведения архивной истории с заданной скоростью. Режим воспроизведения включается из контекстного меню окна графиков. Там же, в контекстном меню можно управлять скоростью воспроизведения, а также интервалом сдвига.

Вспомогательное окно графиков предназначено для анализа схожих ситуаций. Это окно доступно только в продвинутом уровне сложности. Для его отображения следует потянуть за ресайзер, расположенный между тулбаром и основным окном графиков. У вспомогательного окна графиков текущий интервал времени не связан с интервалом времени тренда, что дает возможность сравнивать текущую ситуацию с ситуацией в прошлом.

Работа с легендой и минилегендой

Панель перьев предназначена для отображения списков перьев и информации по ним. Панель перьев имеет несколько представлений, ниже описан детальный вид панели перьев. За дополнительной информацией следует обратиться к справочной части. Детальный вид отображает информацию по каждому перу как на рисунке ниже.



Цвет	Название	Источник	Нижняя граница	Верхняя граница	Размерность	Курсор	Последнее значение	Верх. аз.
<input checked="" type="checkbox"/>	Счетчик	Базовый объект: Файловый архив. Сигнус 1.Смещение	71.1	121.8				
<input checked="" type="checkbox"/>	Значение	Базовый объект: Файловый архив. Сигнус 2.Значение	20.8	200.8	°C	47.76		
<input checked="" type="checkbox"/>	Значение	Базовый объект: Файловый архив. Сигнус 3.Значение	28.39	128.39		65.7		
<input checked="" type="checkbox"/>	Шумовой сигнал	Базовый объект: Файловый архив. Шум Шум.Шумовой с	11.6	111.6	*	88.66		

Элементы управления:

- Чекбокс видимости пера;
- Кнопка вызова *диалога редактирования внешнего вида пера*. Данная кнопка индицирует текущий цвет линий пера;
- Название пера. При клике мыши возможно редактирование названия;
- Источник пера - переменная MasterSCADA;
- Текущие границы вертикального масштаба. При клике мыши возможно редактирование границ;
- Размерность пера;
- Значение пера в срезе *вертикального курсора*;
- Последнее значение пера;
- Кнопки управления видимостью *уставок пера* - заданных границ переменной MasterSCADA;
- Чекбокс управления видимости пера в минилегенде.
- ❖ **Важно!** Если границы переменной не заданы на закладке переменной *Контроль значений* или для шкалы переменной, то кнопка соответствующей *уставки* невидима. Нажатие кнопки включает/выключает видимость заданной *уставки*. При включенной *уставке* – ее значение отображается на кнопке.

При клике мыши на строку с пером – выбранное перо становится *активным пером*. При наведении мыши на строку с пером – оно подсвечивается в *окне графиков*. Также при наведении мыши на перо в окне графиков – оно подсвечивается в панели перьев.

Панель разметки предназначена для отображения списков графических объектов разметки. К графическим объектам разметки относятся все объекты, не являющиеся графиками данных: координатные сетки, оси, уставки, закладки, бордюры, график событий, график состояний. Панель разметки имеет несколько представлений, ниже описан детальный вид.

Детальный вид отображает следующую информацию по каждому элементу разметки:

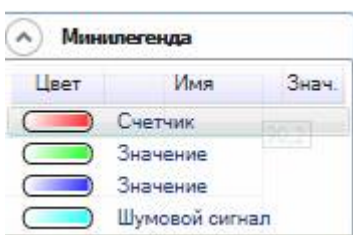
- Чекбокс видимости элемента;
- Кнопка вызова диалога редактирования внешнего вида. Данная кнопка индицирует текущий цвет линий объекта;

- Название объекта. При клике мыши возможно редактирование названия.



При клике мыши на строку с объектом - он становится выбранным. При наведении мыши – подсвечивается в окне графиков. Также при наведении мыши на объект разметки в окне графиков – он подсвечивается в панели разметки.

Минилегенда отображает список перьев. В отличие от панели перьев этот список максимально упрощен. Минилегенда отображает цвет пера, его название и последнее значение. Минилегенда может находиться в свернутом состоянии как показано на рисунке ниже.



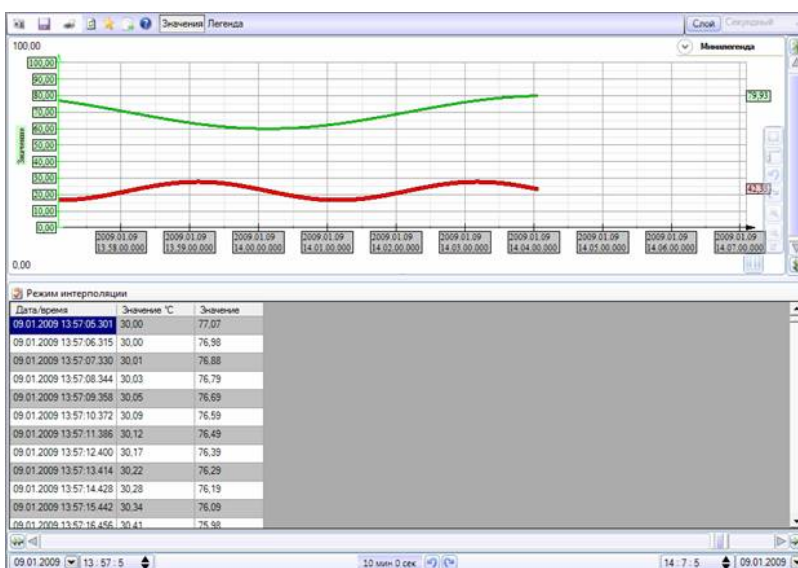
Или в свернутом состоянии, чтобы не мешать просмотру графиков.



Свертывание/развертывание минилегенды осуществляется кликом мыши по шапке легенды. По умолчанию в минилегенды отображаются все перья, которые отображаются в панели перьев, но отображение отдельных перьев в минилегенде (например перьев-уставок) можно запретить из панели перьев. При клике мыши на строку с пером – выбранное перо становится активным пером. При наведении мыши на строку с пером – оно подсвечивается в окне графиков

Отображение данных в табличном виде

В тренде есть возможность отображения данных в табличном виде. Для этого предназначена **Таблица значений**.



Предусмотрен отдельный уровень сложности **Таблица значений**, в котором тренд содержит только эту таблицу. Для вставки таблицы значений на мнемосхему предназначен ActiveX контрол **Таблица значений** в палитре мнемосхемы. Таблица значений отображает данные графиков, видимость которых включена в легенде. Данные отображаются в виде таблицы: в столбцах значения, в строках – время. Для таблицы значений возможно включение режима

интерполяции данных, для этого следует нажать на кнопку с фиксацией *Режим интерполяции*, после чего в таблице будут отображаться интерполированные данные (вычисленные промежуточные значения между значениями, поступившими от источника). Интерполяция для графиков с сегментами *ступенька* (дискретные) производится повторением предыдущего значения. Интерполяция для графиков с сегментами *линия* (аналоговые) производится линейно.

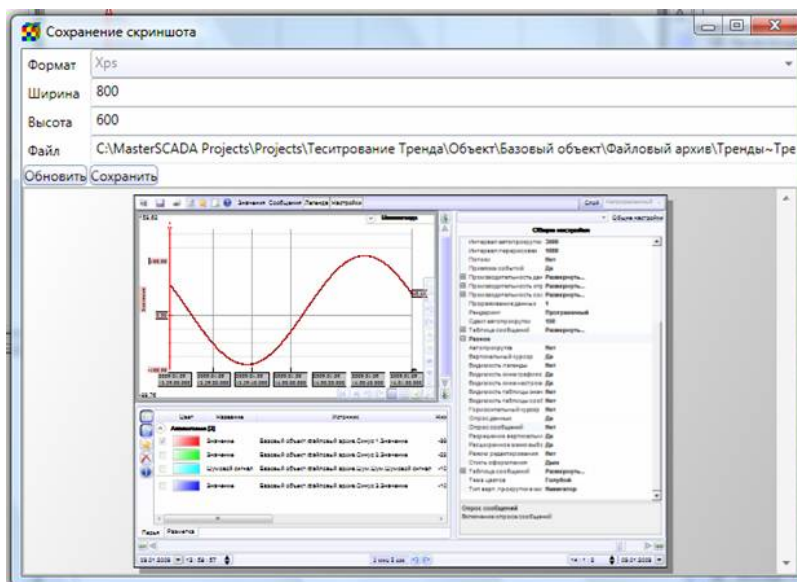
Режим интерполяции		Секунда	x 1
Дата/время	Значение °C	Значение	
09.01.2009 10:57:05.767	30,00	77,03	
09.01.2009 10:57:06.767	30,01	76,94	
09.01.2009 10:57:07.767	30,02	76,84	
09.01.2009 10:57:08.767	30,04	76,75	
09.01.2009 10:57:09.767	30,07	76,65	
09.01.2009 10:57:10.767	30,10	76,55	
09.01.2009 10:57:11.767	30,14	76,46	
09.01.2009 10:57:12.767	30,19	76,36	
09.01.2009 10:57:13.767	30,24	76,26	
09.01.2009 10:57:14.767	30,30	76,16	
09.01.2009 10:57:15.767	30,37	76,05	
09.01.2009 10:57:16.767	30,44	75,95	

Экспорт данных

Экспорт данных производится из тулбара тренда при помощи кнопок:

- Сделать снимок экрана;
- Экспорт;
- Вывести на печать.

Сохранение снимков экрана производится в *диалоговом окне снимков*

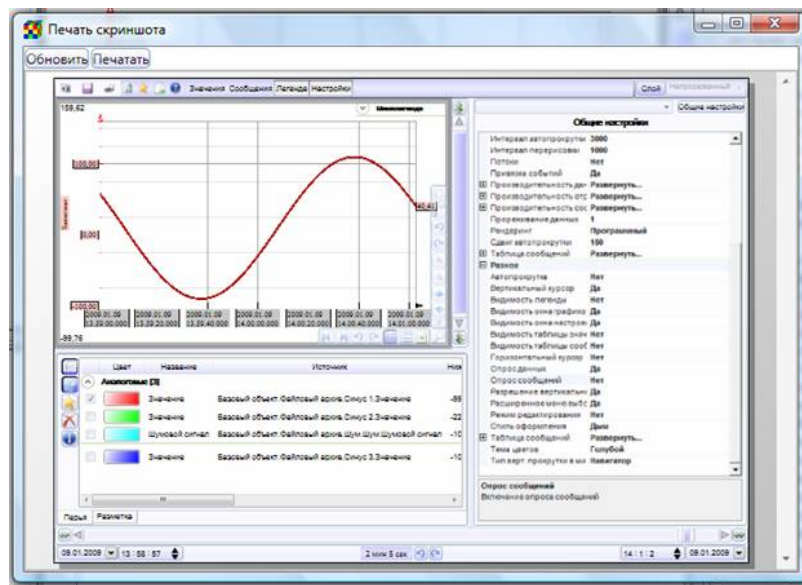


Формат задает формат графического файла для сохранения. Поддерживаются растровые форматы *Bmp, Gif, Gpg, Png, Tiff, Wdp*, а также векторный формат *XPS*, в котором сохраненное изображение может масштабироваться и отображаться без потери качества. Формат *XPS* способен отображаться в последних версиях браузеров. *Ширина* задает разрешение снимка по ширине. *Высота* задает разрешение снимка по высоте. *Трансформация* задает способ масштабирования (масштабирование изменением размеров или изменением разрешения). Данные настройки применимы только для растровых форматов. *Файл* задает файл на диске для сохранения снимка. Кнопка *Обновить* производит обновление зоны предвари-

тельного просмотра. Кнопка *Сохранить* сохраняет снимок в файл с заданными параметрами. Кнопка *Сохранить и закрыть* сохраняет снимок в файл и закрывает диалог.

Сохранение значений в файл производится в формат CSV (формат с разделителями). При сохранении значений в файл выдается стандартное диалоговое окно выбора места сохранения. Сохранение значений выполняется асинхронно и может занимать длительное время, особенно при включенной интерполяции в таблице значений.

Вывод на печать снимков экрана производится в диалоговом окне печати сообщений.



Кнопка *Обновить* производит обновление зоны предварительного просмотра. Кнопка *Печатать* открывает стандартный диалог печати.

Работа с разметкой

Работа с разметкой (настройка элементов разметки окна графиков) производится в *режиме настроек*. В это режиме можно из контекстного меню альтернативным способом добавить:

- Уставки;
- Закладки;
- Задающие графики;
- Графические обработки.

Также из этого контекстного меню можно единственным способом добавить оси и удалить выбранный объект разметки под курсором мыши. В этом меню можно добавить горизонтальную ось значений, которая по умолчанию отсутствует. Эта ось будет в узлах координатной сетки отображать срезы вертикального курсора для обычных графиков, а для графиков зависимости - отображать шкалу графика, отложенного по X.

Работа с задающими графиками


Задающий график – это статичный график, привязанный ко времени, который определяет желаемый вид процесса в будущем. Могут быть и другие применения задающих графиков. Задающий график можно задать двумя способами: таблично и формулой. Добавление задающих графиков производится только в *продвинутом уровне сложности* из контекстного меню *окна графиков* командой *Добавить - Задающий график*. Посмотреть и отредактировать задание задающего графика в любой момент можно в настройках задающего графика *Формула* и *Пользовательская таблица*, а также диалого, который вызывается из тулбара панели перьев по нажатию кнопки *Исходные данные*

Добавление табличных **задающих графиков** производится только в режиме настроек из контекстного меню *окна графиков* командой *Добавить - Задающий график - Задание таблицей*. При выполнении этой команды будет добавлен новый задающий график, а тренд автоматически перейдет в *режим редактирования задатчика*.



Режим редактирования задатчика предназначен для редактирования опорных точек задающего графика. В режиме редактирования задатчика становятся видны задающие точки активного графика (табличные значения), а также изменяется интерфейс мыши:

- Двойной клик по задающей точке удаляет ее с выделенного графика;
- Двойной клик по пустому месту добавляет задающую точку на выделенный график;
- Нажатие кнопки на задающей точке с последующим движением мыши (драг) сдвигает ее, изменяя вид выделенного графика.

Также табличный задающий график добавляется автоматически при переходе в режим редактирования задатчика при помощи кнопки  с панели горизонтального масштаба, если ни один задающий график до сих пор не был создан. Режим редактирования задатчика выключается с панели горизонтального масштаба кнопкой редактирования задатчика или в общих настройках тренда. Кнопка редактирования задатчика видна только в режиме настроек. Для табличного задатчика есть возможность включить зацикливание, при этом заданные значения будут повторяться циклично в будущем. Это можно сделать в настройках задающего графика.

Добавление **задающих графиков**, заданных формулой, производится только в режиме настроек из контекстного меню окна графиков командой *Добавить - Задающий график - Задание формулой*. При выполнении этой команды будет вызван *диалог редактирования формулы* задающего графика. После задания формулы и закрытия диалога – график будет отображен в соответствии с формулой. Задающий график может состоять из нескольких частей и каждая задается своей формулой. Формула определяется как совокупность непосредственно формулы $y = f(t)$ и области ее определения, в которой будет происходить отрисовка. По умолчанию создается график синуса с областью определения равной текущему окну.

Работа с обработками перьев

Обработка пера – это другое перо, данные которого получены из исходных данных трансформацией по некоторому алгоритму. Обработки делятся на:

- Математические: содержат математические операции над пером;
- Фильтры: задают фильтрацию данных пера;
- Операции: задают операции между данными двух перьев.

Добавление обработок, производится из контекстного меню окна графиков. Так как обработка является пером, то на нее может быть также наложена другая обработка, а также для нее применимы все операции с перьями в целом.

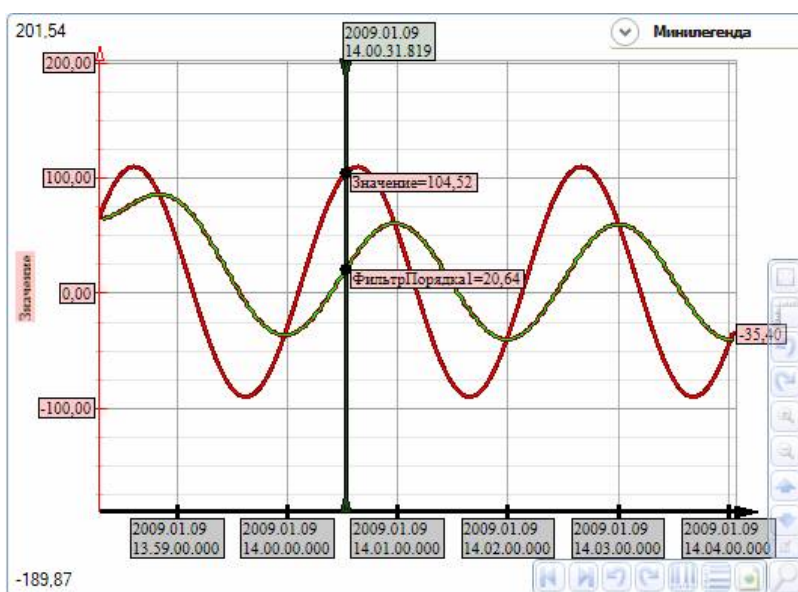
Математические обработки – это обработки, исходными данными которых являются данные одного графика, результатом которых является математическое преобразование исходных данных.

Настройки математической обработки:

- *Интервал*. Определяет кол-во значений, участвующих в групповой обработке. Например, кол-во значений, по которым вычисляется среднее значение (интервал, за который происходит усреднение). Интервал - безразмерная величина, показывает количество исходных значений. Если интервал равен нулю, то он выбирается автоматически как 1/10 количества значений в текущем окне просмотра. Если интервал больше количества значений (например, 1000), обработка рассчитывается для всего окна и отображается в виде константы (например, усреднение всех значений);
- *Действие*. Определяет тип трансформации значений;
- *Аргумент*. Определяет график, трансформация которого осуществляется.

Имеются следующие математические обработки: натуральный логарифм, десятичный логарифм, экспонента, степень двойки, степень десяти, производная, интеграл.

Фильтры – это обработки, исходными данными которых являются данные одного графика, результатом которых являются отфильтрованные исходные данные. Настройки фильтра аналогичны настройкам математических обработок. Имеются следующие фильтры: среднее, экспоненциальное (скользящее) среднее, среднее квадратичное отклонение, дисперсия, фильтр первого порядка, минимальное, максимальное. На рисунке ниже показана обработка *Фильтр первого порядка* для исходного синусоидального графика.



Операции - это обработки, исходными данными которых являются данные двух графиков, а результатом – график операции между этими данными.

Настройки операций:

- *Действие*. Определяет операцию между данными двух графиков;
- *Аргумент X*. Определяет график, являющийся первым аргументом операции;
- *Аргумент Y*. Определяет график, являющийся вторым аргументом операции.

Перечень операций: сумма, разность, произведение, частное, остаток, среднее, зависимость, корреляция. Обратите внимание, что операция *Зависимость* представляет собой обычный график зависимости одного параметра от другого $Y=F(X)$. Если желательно отображение обработок и исходного графика в одинаковых масштабах - следует задать связь с исходным графиком по категории *Окно Y* в настройках.

Наиболее сложной для настройки является операция *Зависимость*. Для добавления зависимости графиков следует перейти в продвинутый вид, режим настроек, затем из контекстного меню выбрать *Добавить операцию - Зависимость графиков*. После этого выбрать созданный график, перейти к его настройкам и установить *Аргумент X* равным графику, откладываемому по X, а *Аргумент Y* равным графику откладываемому по Y.

Работа с динамическими звеньями

Динамические звенья являются разновидностью *задающих графиков*. По сути это и есть задающие графики, для которых задана формула звена.

Добавление звеньев, производится из контекстного меню окна графиков командами *Добавить - Динамические звенья*. По этой команде добавляется график выбранного звена, как кривая разгона, реакция на единичное воздействие. Началу кривой разгона соответствует левая граница времени тренда. Концу кривой разгона – правая граница. Коэффициент усиления звена равен по умолчанию единице. Постоянная времени звена задается как 1/10 от видимого интервала времени. В дальнейшем, границы и настройки звена можно откорректировать в режиме настроек. В тренде определены следующие типы звеньев: апериодическое, двойное апериодическое, пропорциональное, интегрирующее, дифференцирующее, колебательное, консервативное.

У звена есть следующие настройки:

- Начало - начало кривой разгона процесса;
- Конец - конец разгона процесса;
- Тип - тип звена;
- Параметры К, Т.

Работа с трендом в смешанном режиме (тренд + журнал)

Для работы с трендом в смешанном режиме следует в *общих настройках* тренда включить опрос сообщений. После этого становится доступна *таблица сообщений, панель выбора сообщений, панель информации о сообщении*. Также в *окне графиков* происходит отображение сообщений в *графическом виде*, что удобно для анализа возникших в событий при просмотре архива. Все описанные функции являются функциями журнала и описаны в методических указаниях к журналу.

Работа с правами доступа

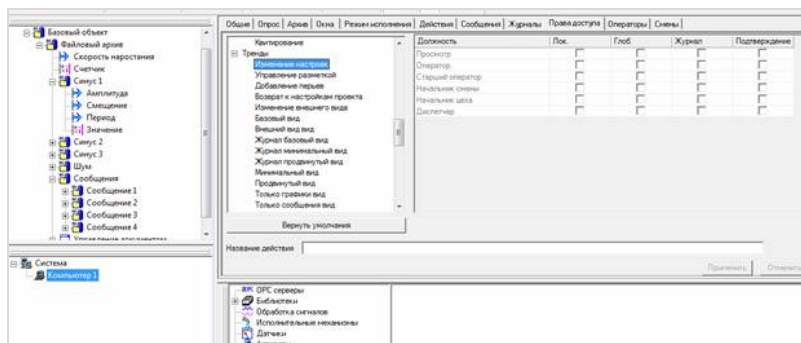
Для тренда определены следующие права доступа оператора:

- *Изменение настроек*. Это право определяет видимость панели настроек, а соответственно возможность редактирования настроек;
- *Управление разметкой*. Право на добавление и удаление элементов разметки окна графиков;
- *Добавление перьев*. Право на добавление и удаление перьев;
- *Возврат к настройкам проекта*. Право на возврат к сохраненным настройкам режима разработки после любых изменений;
- *Изменение внешнего вида*. Право на смену стиля или темы цветов.

Также определены права использования каждого из имеющихся уровней тренда. По умолчанию это следующие права:

- *Базовый уровень*. Право на переключение базового уровня;
- *Продвинутый уровень*. Право на переключение продвинутого уровня;
- *Минимальный уровень*. Право на переключение минимального уровня;
- *Только графики*. Право на переключение уровня *только графики*;
- *Таблица значений*. Право на переключение уровня *таблица значений*.

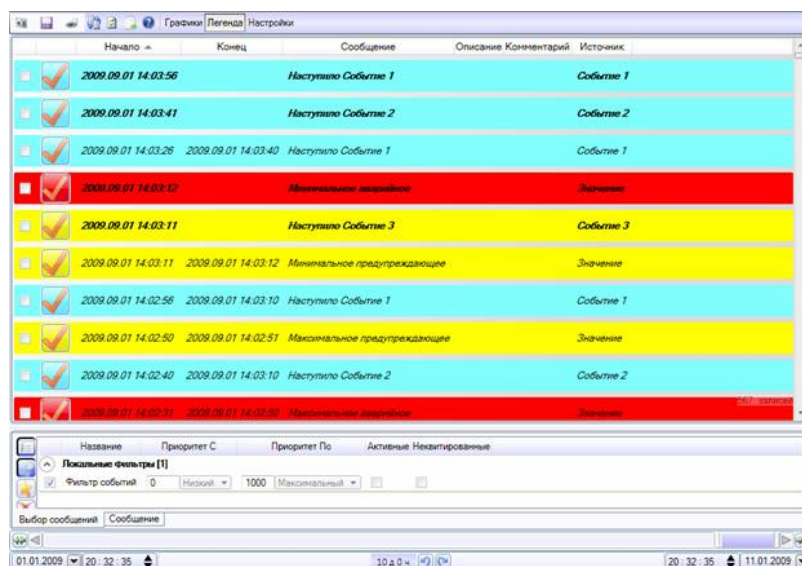
Эти права задаются на уровне узла *Компьютер* дерева системы MasterSCADA для каждого оператора, на закладке *Права доступа*, в категории *Тренды*.



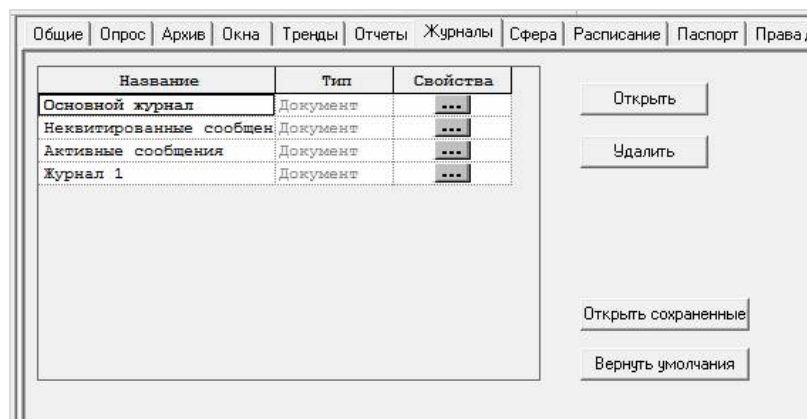
Также на тренд действует общее право для всех документов на печать. Категория *Напечатать документ*. Это право на любые операции экспорта тренда (сохранение в файл и печать).

Методические указания по работе с журналом

Модуль Журнал предназначен для отображения сообщений в табличном и графическом виде с возможностью гибкой фильтрации по различным условиям. Модуль журнал с точки зрения его реализации является одним целым с модулем *Тренд*, поэтому в данных указаниях описана работа только с теми функциями, которые не включены в тренд. Общее описание работы с трендом применимо также и к журналу. При соответствующих правах доступа настройка журнала возможна не только в режиме разработки, но и в режиме исполнения.



Журнал открывается на уровне объекта дерева объектов с закладки *Журналы*.



Также журнал, как и тренд, можно вставить на мнемосхему из палитры как *ActiveX* элемент.

Работа с уровнями сложности журнала аналогична работе с уровнями сложности тренда. Для журнала определены следующие уровни сложности:


- **Продвинутый уровень** является наиболее полным уровнем с возможностью настройки большого количества свойств, предназначен для продвинутых пользователей и детальной настройки. Содержит область графиков, таблицу сообщений, легенду и панель настроек;
- **Базовый уровень** содержит основные визуальные элементы, предназначен для использования без предварительного ознакомления с документацией. Содержит таблицу сообщений, легенду и панель настроек;
- **Минимальный уровень** содержит минимально необходимый набор визуальных элементов для просмотра сообщений, в нем полностью отсутствует возможность настройки, реализован минимум функциональности для неподготовленных пользователей. Содержит только таблицу сообщений;
- **Уровень Только сообщения** содержит только таблицу сообщений;
- **Уровень Только просмотр** содержит только таблицу сообщений без контекстного меню.

Возможно создание пользовательских уровней сложности на основе существующих с помощью специального *редактора уровней сложности*. Для переключения на выбранный уровень - текущий оператор должен иметь соответствующие права. При отсутствии прав - уровень недоступен. Права оператора на переключение уровней сложности задается на уровне компьютера, на закладке *Права доступа*.

На закладке *Журналы* корневого узла дерева системы определяются **Типы журналов**, которые можно использовать в проекте.

- ❖ **Важно!** *Типы журналов, которые редактируются на уровне системы – это не то же самое, что журналы системы. Журналы системы по умолчанию включают в себя все сообщения, возникающие в программе.*

По умолчанию имеются три типа - *журнал активных сообщений*, *журнал неквитированных сообщений* и *основной журнал*, содержащий все сообщения объекта, к которому он относится. Разработчик проекта может добавить любое число дополнительных типов журналов. Типы журналов отличаются друг от друга только названием и предопределенными настройками, прежде всего набором фильтров. Все компьютеры дерева системы и все объекты дерева объектов MasterSCADA могут иметь экземпляры журналы созданных типов, однако их использование становится доступным, только если в режиме разработки зайти в соответствующий журнал и затем выйти из него (изменения в настройках необязательны).

Основное отличие типа журнала от журнала объекта - это возможность задания в качестве источника сообщений *Сообщения текущего объекта*. Если у типа журнала в фильтре стоит настройка *Текущий объект*, то при открытии журнала у *Объекта 1* - в нем будут отображены сообщения *Объекта 1*, а при открытии журнала у *Объекта 2* - в нем будут отображены сообщения *Объекта 2*. Также есть возможность при переопределении настроек журнала объекта - вернуться к настройкам типа журнала при помощи кнопки загрузки настроек типа журнала  в панели управления журналом.

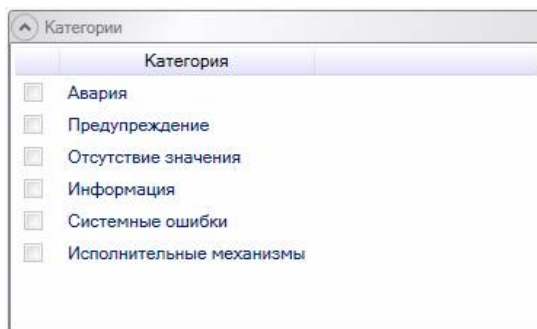
Работа с фильтрами сообщений

Для гибкого и настраиваемого вывода сообщений разработана система фильтрации сообщений. Обратите внимание, что фильтры работают по принципу отбора по условию (а не исключения).

Фильтры сообщений делятся на *локальные* и *глобальные*. Локальные фильтры действуют только на текущий журнал. Глобальные фильтры действуют на все журналы. Глобальные и локальные фильтры сообщений объединяются по *ИЛИ*, то есть журнал содержит сообщения, которые удовлетворяют хотя бы одному из фильтров. Если для журнала не задан ни один фильтр, то сообщения в журнале отсутствуют. При редактировании глобального фильтра - создается его локальная копия и редактирование осуществляется уже для нее. После окончания редактирования эти изменения можно сохранить в глобальном фильтре. Если глобальный фильтр был изменен в режиме исполнения и изменения были сохранены, то при открытии журнала в режиме разработки будет задан вопрос - следует ли применить изменения, сделанные в режиме исполнения или следует их отменить.

Фильтр представляет собой объединение по *И* следующих условий:

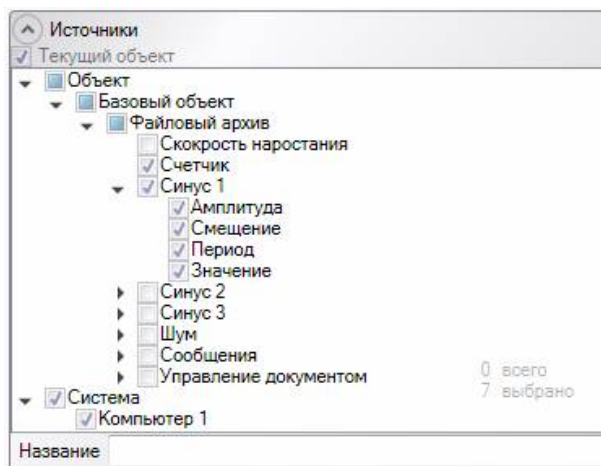
- Подмножество категорий сообщений;



- Диапазон приоритетов;



- Список источников. Список источников состоит из дерева источников и predetermined источника текущий объект. *Текущий объект* – это объект, у которого открыт журнал. При задании данного predetermined источника можно выбрать отображать сообщения всех вложенных объектов или нет (галочка с *подобъектами*). В качестве predetermined источника также можно выбрать "параметры тренда" при использовании журнала в тренде.

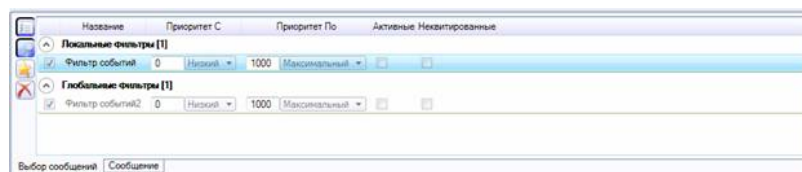


- - состояние сообщений (активность и квитированность)

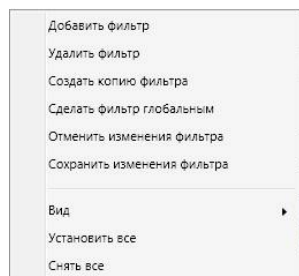


Если фильтр по какой-то группе задан – то заголовок группы подсвечивается синим цветом и справа заголовка появляется кнопка сброса фильтра данной группы в значение по умолчанию.

Панель выбора сообщений содержит список всех фильтров, относящихся к данному журналу.



Контекстное меню панели выбора сообщений содержит следующие команды:

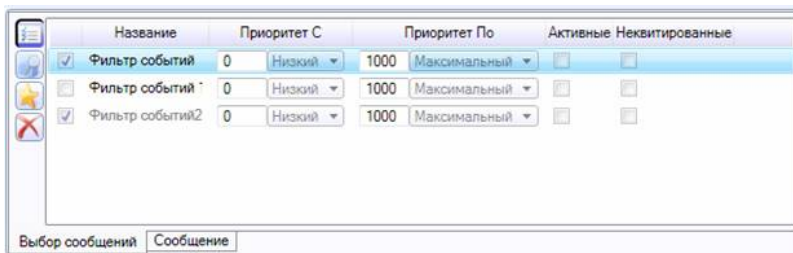


- Добавить фильтр - добавляет локальный фильтр;
- Удалить фильтр - удаляет фильтр;
- Создать копию фильтра - создает локальную копию;
- Сделать фильтр глобальным - делает фильтр глобальным;
- Отменить изменения - отменяет изменения глобального фильтра;
- Сохранить изменения - сохраняет изменения глобального фильтра.

При начале редактирования глобального фильтра - он перемещается в группу *Измененные*, это означает, что он переопределен только для данного журнала и в остальных не изменится. То есть создается локальная копия глобального фильтра, которая скрывает глобальный фильтр. Чтобы применить изменения ко всем журналам (применить изменения к исходному глобальному фильтру) - следует выполнить команду контекстного меню *Сохранить изменения*. Также можно *Отменить изменения*, при этом все изменения глобального фильтра в текущем журнале будут отменены.

Панель выбора фильтров может иметь различные представления. Детальный вид панели отображает следующую информацию по каждому фильтру:

- Чекбокс задействия фильтра;
- Название фильтра. По клику мыши можно переименовать;
- Диапазон приоритетов сообщений;
- Флаг активных сообщений;
- Флаг неквитированных сообщений.

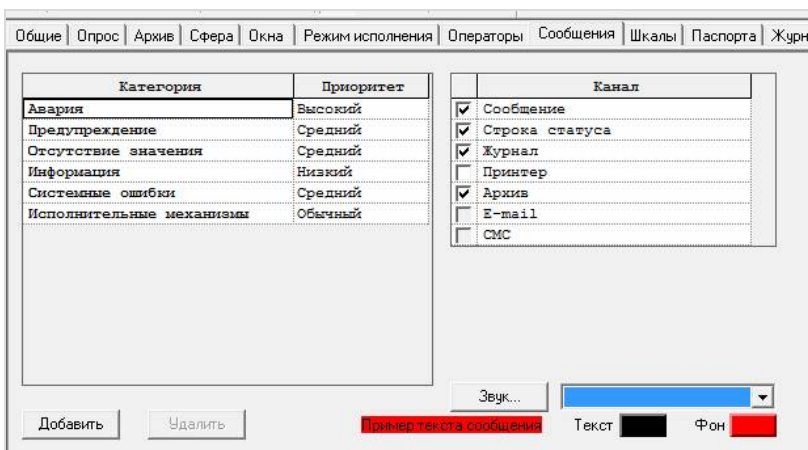


Отображение сообщений

Таблица сообщений служит для отображения списка сообщений в табличном виде.

Начало	Конец	Сообщение	Описание	Комментарий	Источник
2009.09.01 14.03.56		Наступило Событие 1	Событие 1		Событие 1
2009.09.01 14.03.41		Наступило Событие 2	Событие 2		Событие 2
2009.09.01 14.03.28	2009.09.01 14.03.40	Наступило Событие 1	Событие 1		Событие 1
2009.09.01 14.03.12		Максимальное предупреждение	Значение		Значение
2009.09.01 14.03.11		Наступило Событие 3	Событие 3		Событие 3
2009.09.01 14.03.11	2009.09.01 14.03.12	Минимальное предупреждение	Значение		Значение
2009.09.01 14.02.58	2009.09.01 14.03.10	Наступило Событие 1	Событие 1		Событие 1
2009.09.01 14.02.50	2009.09.01 14.02.51	Максимальное предупреждение	Значение		Значение

Неквитированные сообщения (сообщения, прочтение которых оператор не квитировал, то есть не подтвердил) отображаются курсивом, активные (актуальные) сообщения (сообщения, события которых еще не завершены) отображаются жирным шрифтом. Внешний вид сообщения (цвет фона и текста сообщения) задается при определении категории на закладке **Сообщения** узла **Система** дерева системы MasterSCADA.



В таблице сообщений отображается следующая информация:

- Признак пометки сообщений. Служит для временной пометки сообщений с целью дальнейшего анализа;
- Кнопка квитирования сообщения. По нажатию кнопки соответствующее сообщение помечается как квитированное.

Наличие остальных столбцов (полный список соответствует предусмотренному стандартом OPC A&E перечню атрибутов сообщения) не является обязательным. Их видимость задается из контекстного меню *таблицы сообщений*. Видимость может быть включена для следующих столбцов:

- *Начало* - отображает время формирования сообщения;
- *Конец* - отображает время закрытия сообщения;
- *Сообщение* - отображает текст сообщения;
- *Описание* - отображает описание сообщения. Задается на закладке *Общие* у переменной MasterSCADA;
- *Комментарий* - отображает комментарий. Задается на закладке *Общие* у переменной MasterSCADA;
- *Источник* - отображает источник, сформировавший сообщение;
- *Изменения*;
- *Новое состояние* - отображает состояние сообщения;
- *Объект* - объект, содержащий источник сообщения;
- *Последнее квитирование* - время последнего квитирования;
- *Оператор* - оператор, квитировавший сообщение;
- *Тип сообщения* - отображает тип сообщения;
- *Тип источника* - отображает тип источника;
- *Категория* - отображает категорию сообщения;
- *Приоритет* - отображает приоритет сообщения;
- *Условие* - отображает условие формирования сообщения;
- *Подусловие* - отображает подусловие отображения сообщения;
- *Качество* - отображает качество сообщения;
- *Квитирование возможно* - отображает возможность квитирования сообщения;
- *Идентификатор* - уникальный идентификатор сообщения;
- *Примечание* - примечание, введенное оператором во время квитирования.

Панель информации о выбранном сообщении содержит дополнительную информации о текущем сообщении.

Свойство	Значение
Начало	2009.09.01 14:01:09
Конец	2009.09.01 14:02:09
Сообщение	Наступило Событие 3
Описание	
Комментарий	
Источник	Событие 3

Информация разделена на два столбца. Левый столбец содержит название свойства сообщения. Правый - значение свойства. Видимостью свойств можно управлять из контекстного меню, отключая вывод ненужной информации. Перечень свойств тот же самый, что и для таблицы сообщений. Внизу панели выбора сообщений находится текстовое поле для ввода комментария, а также кнопка квитирования сообщения с заданным комментарием.

Отображение сообщений в графическом виде возможно только в уровне сложности *Журнал продвинутый* для журнала и *Тренд продвинутый* для тренда, причем для тренда должен быть включен опрос сообщений в общих настройках. Отображение сообщений в графическом виде производится в *Окне графиков* при помощи графических объектов *График событий* и *Линейный график состояний*.

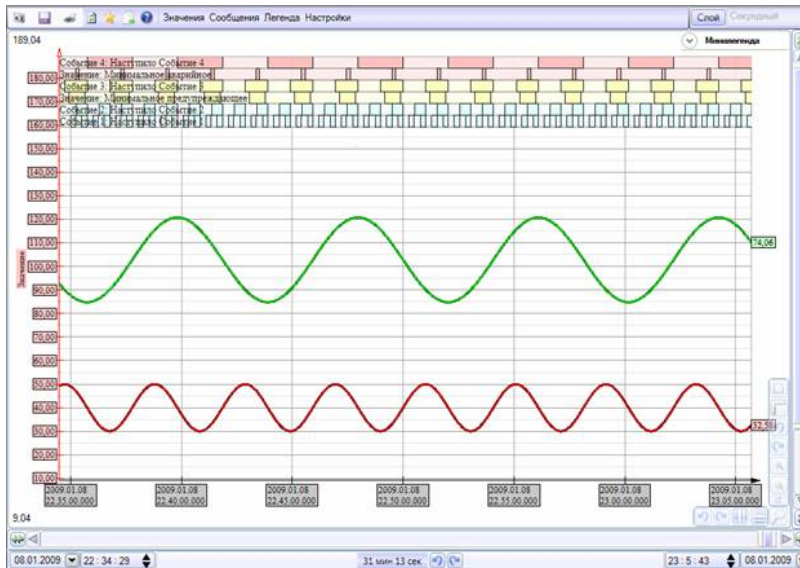
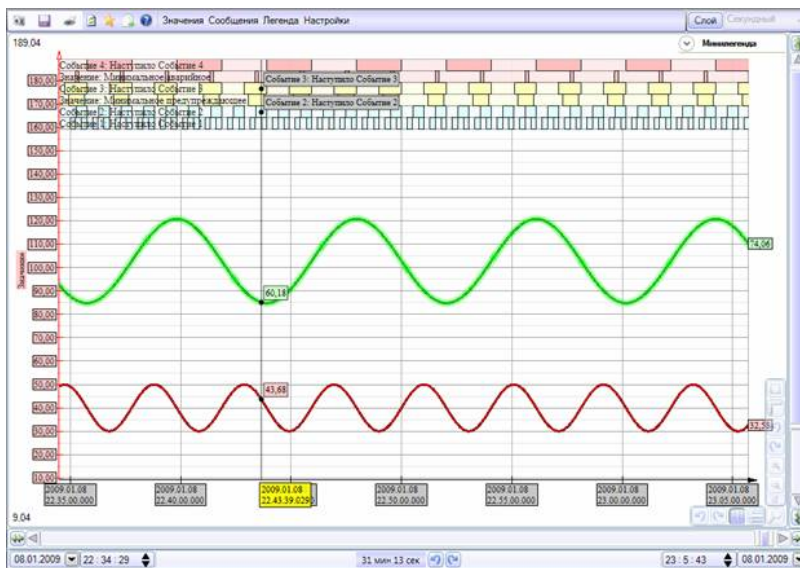


График событий расположен над окном графиков и отображает безусловные сообщения в виде точек. Линейный график состояний отображается в виде прямоугольников, сгруппированных по источникам в порядке убывания приоритета сообщения. Если два события отображаются на экране слишком близко – они сливаются в одно большое событие, это позволяет рассматривать происходящие процессы в разных масштабах времени. В режиме вертикального курсора отображаются срезы линейного графика состояний как показано на картинке ниже.



Экспорт сообщений

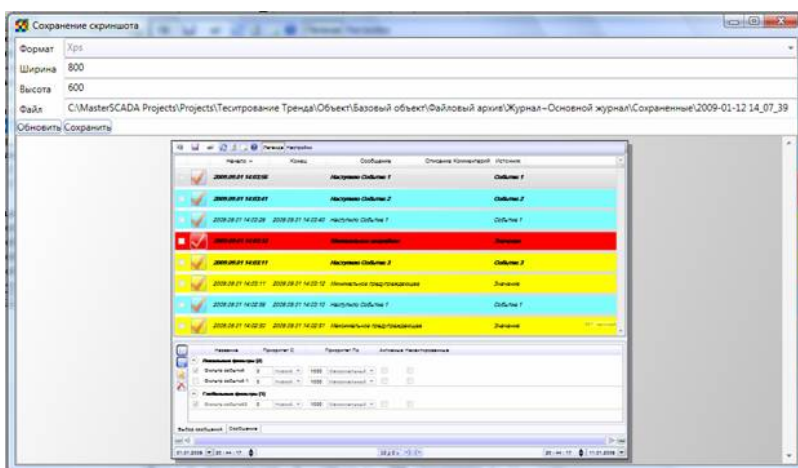
Экспорт сообщений производится из тулбара журнала.



При помощи кнопок:

- Сделать снимок сообщений;
- Экспорт сообщений;
- Вывести на печать.

Сохранение снимков экрана производится в диалоговом окне снимков.



Формат задает формат графического файла для сохранения. Поддерживаются растровые форматы *Bmp, Gif, Gpg, Png, Tiff, Wdp*, а также векторный формат *XPS*, в котором сохраненное изображение может масштабироваться и отображаться без потери качества. Формат *XPS* способен отображаться в браузере. Остальные элементы управления аналогичны подобному окну для трендов.

Сохранение сообщений в файл производится в формат *html*. При сохранении сообщений в файл выдается стандартное диалоговое окно выбора места сохранения. Сохранение сообщений выполняется асинхронно и может занимать длительное время при большом количестве сообщений.

Вывод сообщений на печать производится в *диалоговом окне печати сообщений*. Кнопка *Печать* открывает стандартный диалог печати.



Работа с правами доступа

Для журнала определены следующие права доступа оператора:

- Использование фильтра;
- Настройка фильтра;
- Квитирование;
- Изменения настроек;
- Возврат к настройкам проекта. Право на возврат к сохраненным настройкам режима разработки после любых изменений;
- Изменение внешнего вида. Право на смену стиля или темы цветов, а также на управление столбцами;
- Открытие связанных документов. Право на открытие тренда с источником сообщения, а также любых других, связанных с сообщением документов.

Также определены права использования каждого из имеющихся уровней тренда. По умолчанию это следующие права:

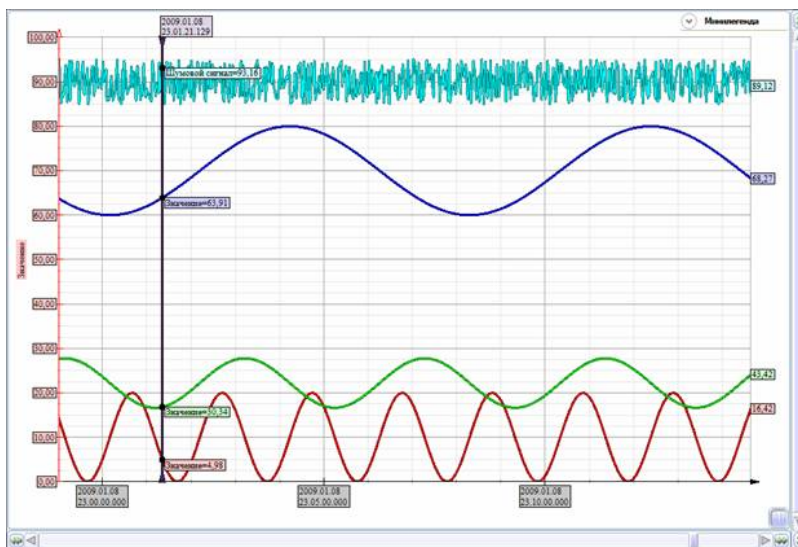
- *Продвинутый уровень.* Право на переключение продвинутого уровня;
- *Базовый уровень.* Право на переключение базового уровня;
- *Минимальный уровень.* Право на переключение минимального уровня;
- *Только сообщения.* Право на переключение уровня *только сообщения*;
- *Только просмотр.* Право на переключение уровня *только просмотр*.

Эти права задаются на уровне узла *Компьютер* дерева системы MasterSCADA для каждого оператора, на закладке *Права доступа*, в категории *Журналы*.

Также на журнал действует общее право для всех документов на печать. Категория *Напечатать документ*. Это право на любые операции экспорта (сохранение в файл и печать).

Справочная часть

Для визуализации графиков зависимости параметров от времени, используется документ типа *Тренд*, который позволяет не только просматривать эти графики в различных масштабах, но и анализировать их.



Для просмотра событий, возникающих в системе, используется документ типа *Журнал*, который позволяет задавать фильтрацию сообщений в различных разрезах.

Начало	Конец	Сообщение	Описание	Комментарий	Источник
2009.09.01 14:03:56		Наступило Событие 1			Событие 1
2009.09.01 14:03:41		Наступило Событие 2			Событие 2
2009.09.01 14:03:26	2009.09.01 14:03:40	Наступило Событие 1			Событие 1
2009.09.01 14:03:12		Максимальное предупреждение			Значение
2009.09.01 14:03:11		Наступило Событие 3			Событие 3
2009.09.01 14:03:11	2009.09.01 14:03:12	Минимальное предупреждение			Значение
2009.09.01 14:02:58	2009.09.01 14:03:10	Наступило Событие 1			Событие 1
2009.09.01 14:02:50	2009.09.01 14:02:51	Максимальное предупреждение			Значение
2009.09.01 14:02:40	2009.09.01 14:03:10	Наступило Событие 2			Событие 2
2009.09.01 14:02:31	2009.09.01 14:02:30	Максимальное предупреждение			Значение

Название	Приоритет С	Приоритет По	Активные	Неактивированные
Локальные фильтры [1]				
Фильтр событий	0	Низкий	1000	Максимальный

Модуль трендов и модуль журналов интегрированы в одном модуле и являются двумя представлениями с предустановленными настройками открытия. Встретив в тексте журнал или тренд следует иметь в виду, что речь идет об одном и том же функциональном модуле. Документ *Тренд* можно полностью настроить как журнал. Документ *Журнал* как тренд настроить нельзя (искусственное ограничение, чтобы сохранить логическую разницу между этими двумя модулями), но можно включить отображение сообщений в графическом виде.

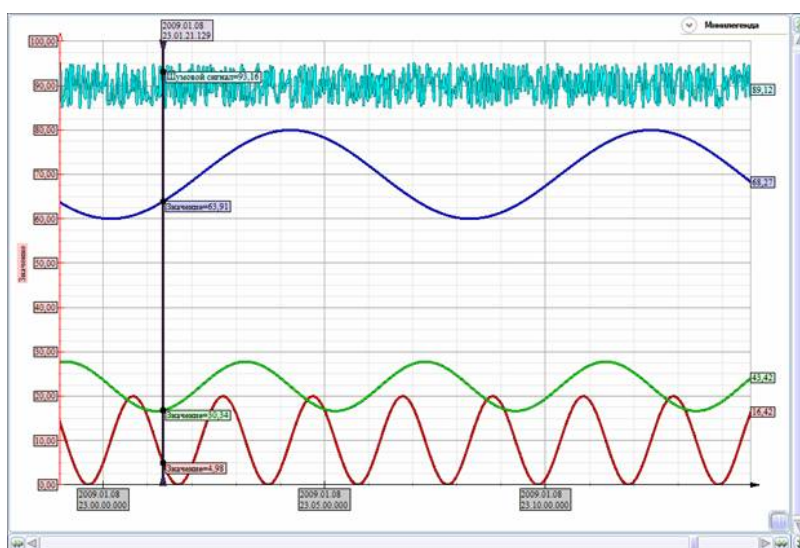
Тренд может быть открыт в нескольких режимах, которые отличаются предустановленными по умолчанию настройками:

- Режим отображения графиков;
- Режим отображения журнала сообщений;
- Режим типа (шаблона) журнала;
- Вставка на мнемосхему графика;
- Вставка на мнемосхему журнала;
- Вставка на мнемосхему таблицы значений.

При этом в любом режиме после открытия могут быть вручную изменены настройки и видимость любых панелей тренда. Например:

- При открытии тренда в режиме графиков можно включить опрос сообщений и отображение сообщений в табличном и графическом виде;
- При открытии тренда в режиме журнала можно включить отображение сообщений в графическом виде.

Тренд в режиме графиков открывается с закладки *Тренды* на уровне объекта MasterSCADA.



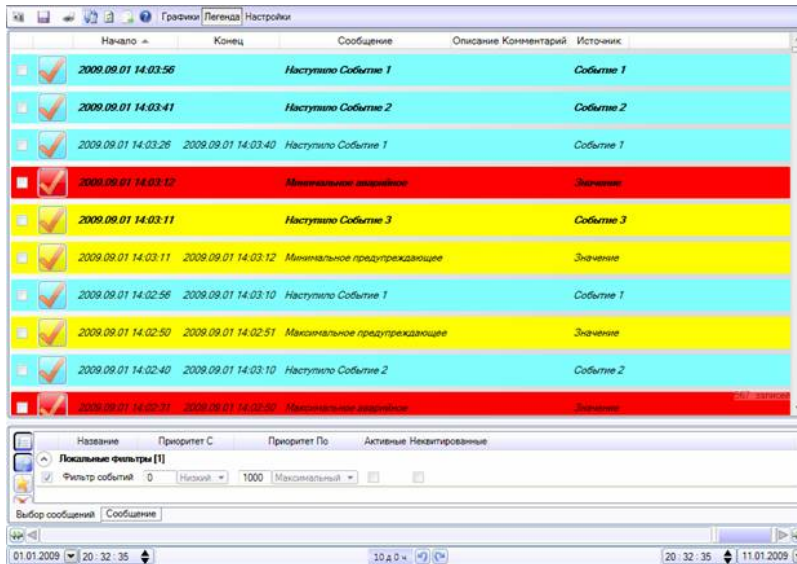
При открытии тренда в режиме графиков по умолчанию выключен опрос сообщений, включен опрос данных, видно окно графиков, в легенде доступна панель перьев, панель разметки. Тренд в режиме графиков содержит следующие характерные для этого режима визуальные элементы:

- Окно графиков;
- Таблица значений;
- Панель перьев в легенде;
- Панель разметки в легенде.

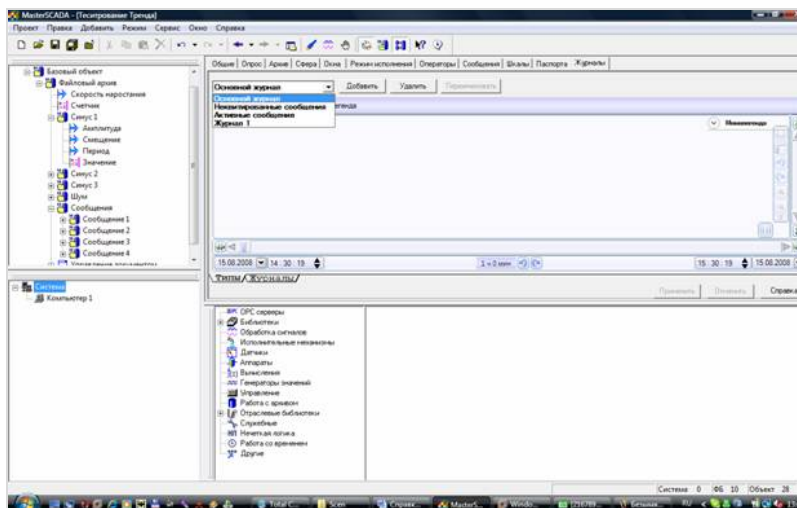
Тренд в режиме журнала открывается с закладки *Журналы* на уровне объекта MasterSCADA. При открытии тренда в режиме журнала по умолчанию включен опрос сообщений, отключен опрос данных, видна таблица сообщений, доступна панель фильтров и панель свойств сообщения.

Тренд в режиме журнала содержит следующие характерные для этого режима визуальные элементы:

- Таблица сообщений;
- Панель сообщений в легенде;
- Панель фильтров в легенде;
- Графическое представление сообщений в окне графиков.



Тренд в режиме типа журнала открывается на закладке MasterSCADA Журналы на уровне узла Система дерева системы MasterSCADA.

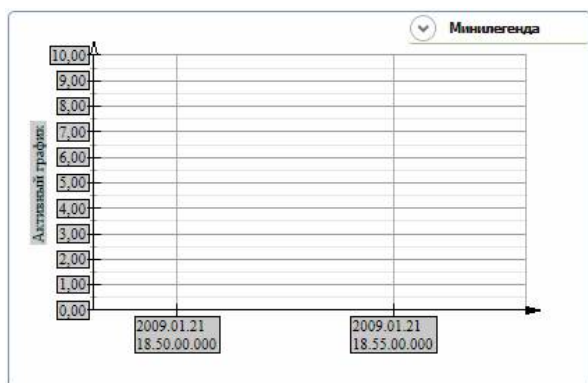


Тип журнала - это предварительно настроенный шаблон журнала объекта. Основное отличие от журнала объекта - это возможность абстрактного задания фильтров сообщений в виде настройки фильтра *Сообщения текущего объекта*. Если у типа журнала в фильтре стоит настройка *Текущий объект*, то при открытии журнала у *Объекта 1* - в нем будут отображены сообщения *Объекта 1*, а при открытии журнала у *Объекта 2* - в нем будут отображены сообщения *Объекта 2*. На основе заданных типов журнала можно открывать журналы объектов без настройки фильтров, например журнал активных сообщений объекта или журнал неактивных сообщений объекта. Также есть возможность при переопределении настроек журнала объекта - вернуться к настройкам типа журнала.

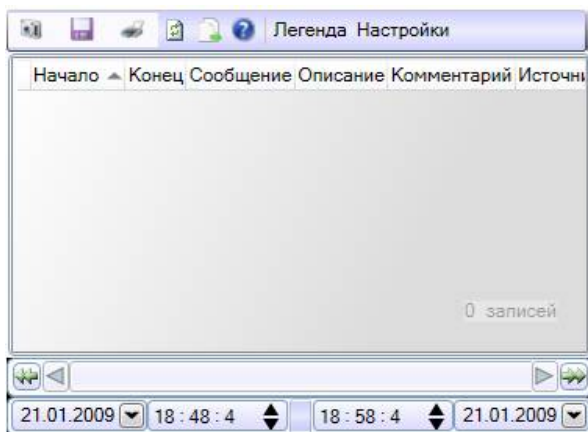
При создании проекта по умолчанию создаются три типа журнала:

- *Активные сообщения* (задан локальный фильтр активных сообщений текущего объекта);
- *Неактивные сообщения* (задан локальный фильтр активных неактивных сообщений текущего объекта);
- *Основной журнал* (задан локальный фильтр всех сообщений текущего объекта).

Вставка на мнемосхему тренда производится из палитры мнемосхемы, из категории *Окно*. При добавлении на мнемосхему устанавливается самый облегченный уровень сложности *Только графики*, в котором отображается только окно графиков. В остальном работа аналогична работе с трендом в виде документа.



Вставка на мнемосхему журнала производится из палитры мнемосхемы, из категории *Окно*. Работа аналогична работе с журналом в виде документа.



Вставка на мнемосхему таблицы значений производится из палитры мнемосхемы, из категории *Окно*. Работа аналогична работе с трендом в виде документа с установленным уровнем сложности *Таблица значений*.

Система данных. Запрос, обработка, выборка, архивация данных происходит в сервере данных. Сервер данных обрабатывает данные асинхронно, что позволяет избежать подвигов системы и замедления реакции тренда на действия пользователя. При работе с данными используются слои данных. *Слои данных* – это заданные интервалы прореживания данных. За счет использования слоев при просмотре истории на больших интервалах времени из архива выбирается значительно меньший (на порядки) объем данных, что снижает нагрузку процессора и ускоряет работу тренда при просмотре архивной истории.

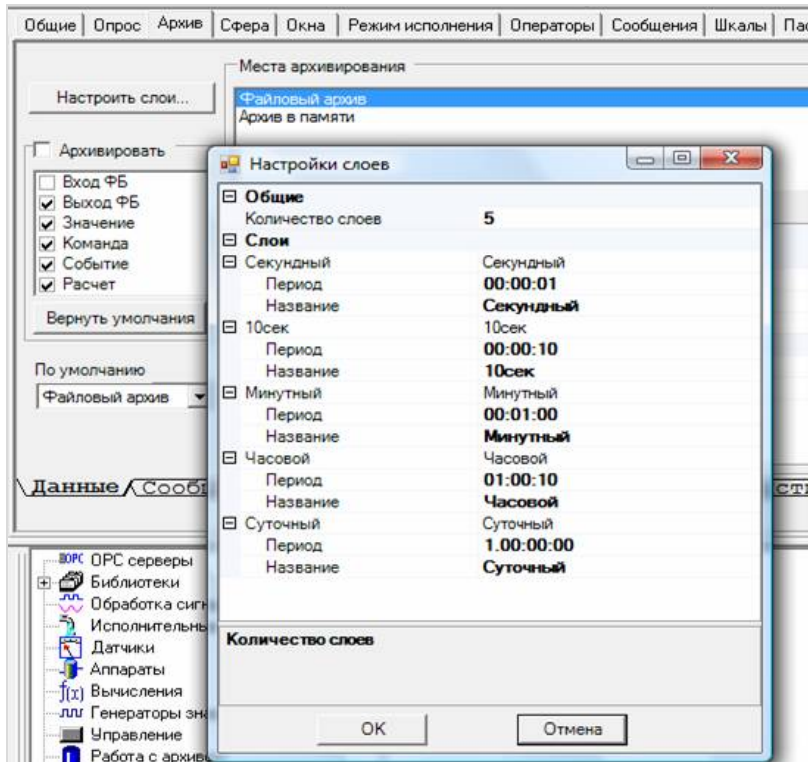
Существуют базовые слои и пользовательские слои. К базовым слоям относятся:

- *Непрореженный слой.* В этом слое хранятся непрореженные данные с теми метками времени, с которыми они поступили в архив;
- *Временный слой.* В этом слое хранятся еще не помещенные в архив данные. Такое может быть если для архивирования используется отдельный сервер и связь с ним временно отсутствует.

По умолчанию для нового проекта создаются следующие пользовательские слои:

- *Минутный слой.* Хранит не более 4 значений за каждую минуту: первое, последнее, максимальное, минимальное;
- *Часовой слой.* Хранит не более 4 значений за каждый час;
- *Суточный слой.* Хранит не более 4 значений за каждые 24 часа.

В проекте может быть настроено неограниченное количество пользовательских слоев. Настройка производится на уровне узла *Система* дерева системы MasterSCADA с закладки *Архив* кнопкой *Настроить слои*.



В файловом архиве слои данных не реализованы, а реализованы только для архивов баз данных. Поэтому хранение архивов в базе данных предпочтительней чем в файловых, если предполагается просматривать историю за большие периоды времени. Разница становится заметной, если кол-во значений переменной за просматриваемый период больше разрешения экрана по горизонтали в несколько раз. Например, если переменная опрашивается раз в секунду, то за месяц из файлового архива будет запрошено 2,6 млн. значений, тогда как из архива базы данных с использованием слоев будет запрошено значительно меньше значений, не больше 40-50 тысяч в зависимости от настройки слоев данных.

Система сообщений. Запрос, обработка, выборка, архивация сообщений происходит в сервере сообщений. Сервер сообщений обрабатывает сообщения асинхронно, что позволяет избежать подвисаний системы и замедления реакции тренда на действия пользователя. Для гибкого и настраиваемого вывода сообщений разработана система фильтрации сообщений. Фильтры работают по принципу отбора по условию (а не по исключению).

Фильтры сообщений делятся на локальные и глобальные:

- *Локальные фильтры* действуют только на текущий журнал;
- *Глобальные фильтры* действуют на все журналы.

Глобальные и локальные фильтры сообщений объединяются по *ИЛИ*, то есть журнал содержит сообщения, которые удовлетворяют хотя бы одному из фильтров. Если для журнала не задан ни один фильтр, то сообщения в журнале отсутствуют. При начале редактирования глобального фильтра - он перемещается в группу *Измененные*, это означает, что он переопределен только для данного журнала и в остальных не изменится. То есть создается локальная копия глобального фильтра, которая скрывает глобальный фильтр. Глобальные фильтры можно переопределять на уровне текущего документа, при этом изменения будут применены только к этому документу. Применить изменения к глобальному фильтру (ко всем документам) можно вручную.

Общие свойства Тренда – отображают свойства всего модуля в целом, влияющие на получение, обработку, отрисовку данных. Общие свойства можно просматривать и изменять в панели настроек.

Категория **Данные**:

- *Максимум графиков* – максимальное количество графиков, которое можно разместить на тренде;
- *Название пера* – определяет название новых перьев, добавляемых на тренд. Не влияет на названия уже добавленных перьев (тип названия уже добавленного пера следует изменять из свойств пера);

- *Настройки печати* – настройки, влияющие на печать тренда. Настройка ориентация определяет ориентацию альбомного листа при печати (ориентацию можно поменять также в настройках принтера в диалоге печати). Настройка *Видимость легенды* влияет на видимость легенды при печати для интернет-клиента. В настройках печати можно настроить черно-белую печать;
- *Настройки сохранения* – настройки, влияющие на сохранение скриншота тренда. Настройка *Формат файла* влияет на формат сохраняемого файла;
- *Настройка Высота и Ширина* задают размер сохраняемого изображения;
- *Настройка Трансформация* определяет способ масштабирования, если текущий размер тренда на экране отличается от заданного размера скриншота. Трансформация *размер* определяет трансформацию таким образом, чтобы размер шрифтов остался неизменным в абсолютных величинах, однако в этом случае относительно нового размера изображения шрифты будут мельче. Трансформация *разрешение* определяет трансформацию таким образом, чтобы размер шрифтов остался неизменным относительно нового размера, однако абсолютный размер шрифтов при этом изменится;
- *Настройки таблицы значений* - настройки функциональности и внешнего вида таблицы значений;
- *Настройка Базовый период интерполяции* - базовый временной период используемый при интерполяции значений;
- *Настройка Вид колонок дата/время* влияет на внешний вид соответствующих столбцов;
- Настройка *Вывод в цвете* влияет на заливку таблицы значений серым цветом;
- *Настройка Коэффициент к базовому периоду интерполяции* - коэффициент, на который умножается базовый временной период интерполяции, результат является шагом интерполяции значений;
- *Настройки Период С и Период По* задают левую и правую временные границы таблицы значений;
- Настройка *Режим интерполяции* включает/отключает режим интерполяции, при включенном режиме в таблице значений отображаются реально имеющиеся данные, при включенном режиме в таблице значений отображаются интерполированные данные;
- *Настройка Формат времени* влияет на формат времени, отображаемого в таблице значений;
- *Настройка Формат даты* влияет на формат даты, отображаемой в таблице значений;
- *Настройка Разделитель CSV* определяет знак, использующийся в качестве разделителя при сохранении табличных значений в файл.

Настройки *Высота*, *Ширина*, *Трансформация* применяются только для растровых форматов файлов (*bmp*, *gif*, *jpg*, *png*, *tif*, *wdp*). При сохранении в векторный формат они не имеют смысла.

Категория **Окно**:

- Левая граница – задает левую временную границу тренда;
- Правая граница – задает правую временную границу тренда;
- Связанный тренд – задает тренд, временные границы которого связаны с данным. В выпадающем списке отображаются все тренды, открытые на данный момент.

Категория **Производительность/Качество**:

- *Антиалиасинг графики* - при включенной настройке улучшается качество отрисовки графики, происходит сглаживание линий. Производительность уменьшается при действии;
- *Видимость мини-сетки* - при изменении настройки включается/отключается видимость вспомогательной координатной сетки. Если вспомогательная сетка отключена производительность увеличивается, так как происходит отрисовка меньшего количества линий;
- *Групповая производительность* - это группа предопределенных настроек данной категории, которые можно переключать из одного места, если не нужна тонкая настройка;
- *Интервал обновления данных* - задает период перерисовки данных на экране. Задается в миллисекундах. При увеличении значения в *n* раз – загрузка процессора трендом снизится в *n* раз;

- *Интерфейс мыши.* Включает/отключает интерфейс мыши для окна графиков. При этом интерфейс мыши для остальных визуальных элементов не отключается;
- *Качественный курсор.* Задействует улучшенную отрисовку графического курсора, что ведет к снижению производительности. Улучшенная отрисовка заключается в том, что срезы курсора отображаются во время движения мыши, а не только когда она остановится;
- *Отображать FPS.* Задействует вывод дополнительной отладочной информации. Производительность уменьшается;
- *Период синхронизации.* Применяется вместе с настройкой *Синхронное обновление*. Когда включено синхронное обновление, то при сдвиге окна графиков мышью – все остальные визуальные элементы, например *Календарь* или *Время* обновляются моментально. Если синхронное обновление отключено – то визуальные элементы обновляются через заданный период после остановки мыши. Синхронное обновление увеличивает качество отрисовки и уменьшает производительность;
- *Подсветка мышью.* Задействует подсветку графического объекта под курсором мыши. При включенной подсветке производительность уменьшается;
- *Привязка событий.* Задействует графическую привязку безусловных сообщений к источнику сообщения. То есть сообщение от источника *Температура1* будет располагаться на экране на графике *Температура1*. При выключенной настройке это сообщение будет располагаться вверху экрана. Задействование настройки снижает производительность;
- *Прореживание данных.* Задается в пикселях и показывает как часто надо прореживать данные. Если значение равно нулю – прореживание отключено и все имеющиеся значения будут отображены на экране (даже если их сотни тысяч). Если значение равно единице – прореживание осуществляется так, чтобы в каждом пикселе было не больше одного значения, при этом на экране будет отображено 1000-2000 значений при любом количестве исходных значений. Отключение прореживания сильно ухудшает производительность. Увеличение шага прореживания увеличивает производительность, но ухудшает правильность отображения внешнего вида графика;
- *Рендеринг* переключает тип рендеринга. Программный рендеринг осуществляется процессором, аппаратный осуществляется видеокартой. На слабых видеокартах программный рендеринг предпочтительней;
- *Сдвиг автопрокрутки* задается в пикселях. Если сдвиг автопрокрутки равен нулю – то в режиме автопрокрутки окно графиков сдвигается при каждой перерисовке данных. Иначе происходит разовый дискретный сдвиг окна графиков на указанное значение, а затем графики перерисовываются без сдвига, пока не дойдут до края экрана;
- *Тултип графиков.* При задействовании отображается тултип графика под курсором мыши.

Категория **Разное**:

- *Опрос данных* - включает опрос данных;
- *Опрос сообщений* - включает опрос сообщений;
- *Тип верт. прокрутки в мин. режиме.*

Категория Таблицы:

- *Таблица сообщений* – настройки таблицы сообщений;
- *Таблица значений* – настройки таблицы значений;
- *Таблица перьев* – настройки таблицы перьев;
- *Таблица разметки* – настройки таблицы разметки;
- *Таблица фильтров* – настройки таблицы фильтров;
- *Минилегенда* – настройки таблицы минилегенды;

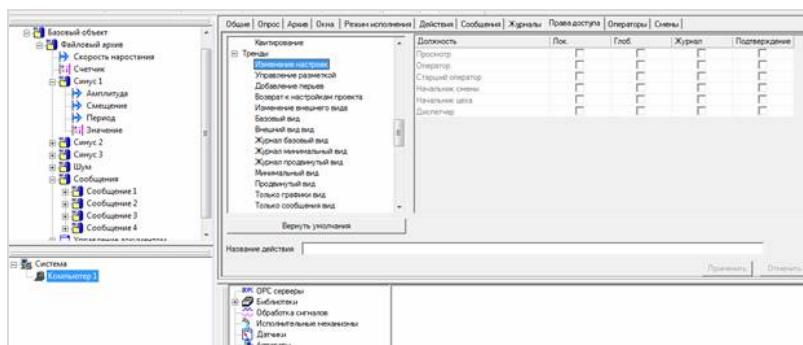
Для тренда определены следующие **права доступа оператора**:

- *Изменение настроек.* Это право определяет видимость панели настроек, а соответственно возможность редактирования настроек;
- *Управление разметкой.* Право на добавление и удаление элементов разметки окна графиков;
- *Добавление перьев.* Право на добавление и удаление перьев;
- *Возврат к настройкам проекта.* Право на возврат к сохраненным настройкам режима разработки после любых изменений;
- *Изменение внешнего вида.* Право на смену стиля или темы цветов.

Также определены права использования каждого из имеющихся уровней тренда. По умолчанию это следующие права:

- *Базовый уровень*. Право на переключение базового уровня;
- *Продвинутый уровень*. Право на переключение продвинутого уровня;
- *Минимальный уровень*. Право на переключение минимального уровня;
- *Только графики*. Право на переключение уровня *Только графики*;
- *Таблица значений*. Право на переключение уровня *Таблица значений*;

Эти права задаются на уровне узла *Компьютер* дерева системы MasterSCADA для каждого оператора, на закладке *Права доступа*, в категории *Тренды*.



Также на тренд действует общее право для всех документов на печать. Категория *Напечатать документ*. Это право на любые операции экспорта тренда (сохранение в файл и печать).

Для журнала определены следующие **права доступа оператора**:

- *Использование фильтра*;
- *Настройка фильтра*;
- *Квитирование*;
- *Изменения настроек*;
- *Возврат к настройкам проекта*. Право на возврат к сохраненным настройкам режима разработки после любых изменений;
- *Изменение внешнего вида*. Право на смену стиля или темы цветов, а также на управление столбцами;
- *Открытие связанных документов*. Право на открытие тренда с источником сообщения, а также любых других, связанных с сообщением документов.

Также определены права использования каждого из имеющихся уровней тренда. По умолчанию:

- *Продвинутый уровень*. Право на переключение продвинутого уровня;
- *Базовый уровень*. Право на переключение базового уровня;
- *Минимальный уровень*. Право на переключение минимального уровня;
- *Только сообщения*. Право на переключение уровня *Только сообщения*;
- *Только просмотр*. Право на переключение уровня *Только просмотр*.

Эти права задаются на уровне узла *Компьютер* дерева системы MasterSCADA для каждого оператора, на закладке *Права доступа*, в категории *Журналы*. Также на журнал действует общее право для всех документов на печать. Категория *Напечатать документ*. Это право на любые операции экспорта (сохранение в файл и печать).

В тренде предусмотрена возможность **экспорта данных и сообщений**. К экспорту относится сохранением снимков экрана, экспорт данных и сообщений в файл, вывод на печать. Сохранение снимков экрана производится в *диалоговом окне снимков*. Экспорт снимка экрана возможен в следующие растровые форматы: *Bmp, Gif, Gpg, Png, Tiff, Wdp*, а также векторный формат *XPS*, в котором сохраненное изображение может масштабироваться и отображаться без потери качества. При экспорте снимка экрана в растровые форматы возможно задание разрешения снимка. Векторный формат *XPS* можно просмотреть в браузере, а также распечатать из браузера. Экспорт значений и сообщений возможен в следующие форматы: *CSV, HTML*. При экспорте в *CSV* задается разделитель значений. Вывод снимков экрана на печать производится в диалоговом окне печати снимков. Вывод на принтер осуществляется аналогично экспорту в файл. Возможна печать любого документа на виртуальный *XPS*

принтер (создается файл формата *XPS* с возможностью дальнейшего просмотра и печати из браузера).

Внешний вид

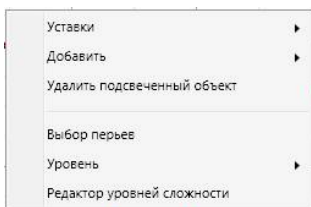
Внешний вид документа определяется уровнем сложности интерфейса, стилем и темой цветов. Любой из этих компонентов можно переключать независимо. Уровень сложности определяет наличие визуальных элементов, стиль – их внешность, тема цветов – цвет. Возможность изменения внешнего вида задается правами доступа оператора. *Уровнем сложности* является совокупность всех визуальных элементов документа (визуальное дерево), а также свойств (дерево свойств). Уровни тренда можно переключать, редактировать, переопределять.

Уровни делятся на шаблонные (уровни сложности, поставляемые в комплекте с MasterSCADA) и пользовательские (настраиваемые уровни проекта). В комплекте поставляется несколько предустановленных шаблонных уровней сложности. Пользовательские уровни сложности по умолчанию отсутствуют. При создании пользовательских уровней они создаются в проекте наследованием от уже существующих уровней (шаблонных или пользовательских). Если название пользовательского и шаблонного уровня MasterSCADA совпадают, то шаблонный уровень перекрывается пользовательским. Уровень сложности определяет наличие и тип визуальных элементов (интерфейс пользователя). Два разных уровня сложности могут содержать разные визуальные элементы. Редактирование уровней сложности осуществляется в редакторе уровней. Для переключения на выбранный уровень - текущий оператор должен иметь соответствующие права доступа к этому уровню сложности.

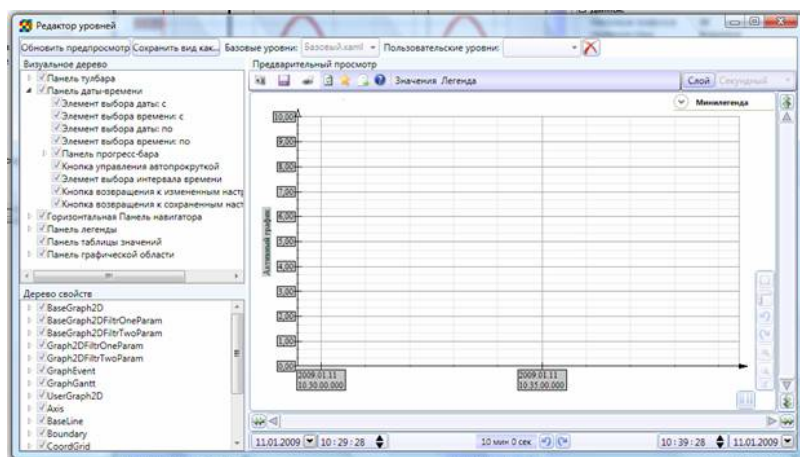
В комплекте MasterSCADA поставляются следующие уровни сложности тренда в порядке увеличения сложности: *только графики*, *минимальный*, *базовый*, *продвинутый*. Для журнала соответственно: *только сообщения*, *журнал минимальный*, *журнал базовый*, *журнал продвинутый*. Использование этих уровней произвольно, но изначальное предназначение следующее:

- *Продвинутый уровень* является наиболее полным видом с возможностью настройки большого кол-ва свойств, предназначен для продвинутых пользователей и детальной настройки;
- *Базовый уровень* содержит только основные визуальные элементы, предназначен для использования без предварительного ознакомления с документацией, интерфейс пользователя интуитивно понятен;
- *Минимальный уровень* содержит необходимый минимум визуальных элементов для просмотра графиков, в нем полностью отсутствует возможность настройки, реализован минимум функциональности для неподготовленных пользователей;
- *Уровень Только графики/Только сообщения* предназначен для использования на мнемосхемах, содержит только область для графического представления перьев или только таблицу сообщений.

Редактор уровней сложности вызывается из контекстного меню окна графиков в режиме настроек.



Создание нового уровня осуществляется наследованием от одного из существующих уровней (шаблонных или пользовательских).



Редактирование включает следующие шаги:

- **Выбор существующего уровня.** При этом он будет показан в окне предварительного просмотра;
- **Выбор видимых визуальных элементов уровня.** По умолчанию все визуальные элементы видимы. Сделать элемент невидимым можно, сняв галочку напротив его названия в визуальном дереве;
- **Выбор свойств графических объектов** (свойств, отображаемых в панели настроек для выделенного объекта). Сделать свойство невидимым можно, сняв галочку напротив его названия в визуальном дереве;
- **Предварительный просмотр результата** по нажатию кнопки *Обновить предпросмотр*;
- **Сохранение вида** по нажатию кнопки *Сохранить вид как*. Если название пользовательского вида совпадет с названием шаблонного вида, то он перекроет шаблонный (шаблонный вид MasterSCADA останется, но не будет загружаться и отображаться как доступный для оператора)

Удаление пользовательского уровня включает следующие шаги:

- **Выбор существующего пользовательского уровня.** При этом он будет показан в окне предварительного просмотра;
- **Удаление пользовательского уровня** кнопкой *Удалить уровень проекта*.

Уровни тренда разработаны на языке структурной разметки XAML. В большинстве случаев шаблонных видов и пользовательских видов, наследуемых от них достаточно, но при желании можно разработать полностью свой вид на языке структурной разметки XAML (в текстовом редакторе или любом редакторе XAML, например *Microsoft Expression Blend*). При разработке своего вида на языке XAML возможно создание своего пользовательского интерфейса, отличного от описанного здесь. Для этого требуется хорошее знание языка структурной разметки XAML, а также документация разработчика для модуля тренд.

Стили определяют следующие характеристики внешнего вида документа:

- Градиент раскраски элементов;
- Анимация элементов;
- Реакция элемента на мышшь.

В тренде изначально определено три стиля: *дым*, *стекло* и *классический*. Стиль дым оформляет визуальные элементы полупрозрачностью. Стиль стекло оформляет в контрастной гамме двух базовых цветов. Стиль классический оформляет в зависимости от текущей темы операционной системы. Пользователи с хорошим знанием XAML могут разработать стиль самостоятельно. Для этого следует обратиться к документации разработчика модуля тренд. Для переключения стилей текущий оператор должен иметь права доступа на смену оформления внешнего вида. При отсутствии прав смена стилей невозможна.

Тема цветов определяет цвета визуальных элементов документа. В тренде изначально определены следующие цветовые темы: голубая, розовая, салатная, оранжевая, черно-белая. Пользователи с хорошим знанием XAML могут разработать тему цветов самостоятельно. Для этого следует обратиться к документации разработчика модуля тренд. Для переключения цветовых тем - текущий оператор должен иметь права доступа на смену оформления внешнего вида. При отсутствии прав смена цветовых тем невозможна.

Панели

Панели - это визуальные части тренда, содержащие сгруппированные логически объекты. Видимостью панелей можно управлять. В разных уровнях сложности доступность панелей может меняться. Интерфейс пользователя в этом разделе описан для уровня сложности *Продвинутый*, на основе которого унаследованы все шаблонные и пользовательские виды. Следует учесть, что в уровне *Базовый*, *Минимальный* и *Только графики* изменять настройки вообще нельзя, а в пользовательских уровнях видимость настроек для изменения задается произвольно.

При разработке своего уровня сложности на языке *XAML* возможно создание своего пользовательского интерфейса, отличного от описанного здесь (при этом перечень свойств и настроек останется неизменным). Для разработки своего уровня требуется хорошее знание языка структурного описания разметки *XAML*, а также документация программиста для модуля тренд. В зависимости от некоторых общих настроек тренда - отдельные элементы могут быть не видны.

Задействование опроса данных влияет на видимость:

- Кнопка тулбара *Значения*;
- Кнопка тулбара *Графики*;
- Окно графиков;
- Панель значений.

Задействование опроса сообщений влияет на видимость:

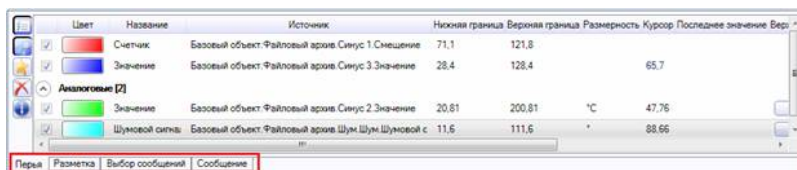
- Кнопка *Сообщения*;
- Таблица сообщений;
- Панель фильтров;
- Панель детальной информации сообщения.

Задействование автопрокрутки влияет на видимость:

- Элементов *Календарь*;
- Элементов *Время*;
- Навигатора.

Панель легенды является контейнером для дополнительных информационных панелей, таких как:

- Панель перьев;
- Панель разметки;
- Панель выбора сообщений;
- Панель расширенной информации о сообщении.



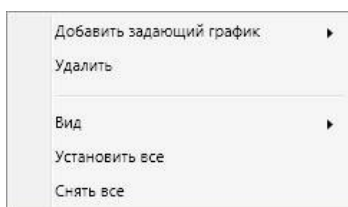
Цвет	Название	Источник	Нижняя граница	Верхняя граница	Размерность	Курсор	Последнее значение	Верг.
Счетчик	Базовый объект: Файловый архив: Синус 1.Смещение	71,1	121,8					
Значение	Базовый объект: Файловый архив: Синус 3.Значение	28,4	128,4	65,7				
Аналоговые [2]	Значение	Базовый объект: Файловый архив: Синус 2.Значение	20,81	200,81	°C	47,76		
Шумовой сигнал	Базовый объект: Файловый архив: Шум Шум Шумовой с.	11,6	111,6	*		88,66		

Панель легенды может быть видима или скрыта с экрана. Видимость панели управляется из тулбара *Тренда*

Панель перьев, левая панель рисунка выше, отображает список всех перьев тренда (переменных MasterSCADA, задающих графиков и графических обработок). Тулбар *Панели перьев* содержит следующие кнопки, сверху вниз:

- *Переключение вида* - простой/детальный;
- *Группировка перьев* - группировка по шкале/группировка по типу пера/группировка отключена. Переключение группировки осуществляется последовательными кликами мыши по кнопке. Включенное состояние группировки индицируется утопленной кнопкой;
- *Добавить перья* - вызывает диалог *Выбора перьев*;
- *Удалить выделенное перо* - удаляет активное перо;
- *Открыть диалог редактирования таблицы/формулы пера* - открывается соответствующий диалог.

Контекстное меню панели перьев содержит следующие команды:



- *Добавить задающий график.* Добавляет Задающий график;
- *Удалить.* Удаляет выделенные перья;
- *Вид: простой/детальный.* Устанавливает простой или детальный вид;
- *Показать все.* Делает все перья видимыми;
- *Скрыть все.* Делает все перья невидимыми.

Вид панели перьев может быть простой или детальный. Простой вид отображает следующую информацию по каждому перу:

- Чекбокс видимости пера;
- Название пера.



Детальный вид отображает следующую информацию по каждому перу:

Цвет	Название	Источник	Нижняя граница	Верхняя граница	Размерность	Курсор	Последнее значение	Верх
<input checked="" type="checkbox"/>	Счетчик	Базовый объект: Файловый архив: Сигнус 1. Сместение	71,1	121,8				
<input checked="" type="checkbox"/>	Значение	Базовый объект: Файловый архив: Сигнус 2. Значение	20,8	200,8	°C		47,76	
<input checked="" type="checkbox"/>	Значение	Базовый объект: Файловый архив: Сигнус 3. Значение	28,39	128,39			65,7	
<input checked="" type="checkbox"/>	Шумовой сигнал	Базовый объект: Файловый архив: Шум Шум. Шумовой с	11,6	111,6	*		88,66	

- Чекбокс видимости пера;
- Кнопка вызова *Диалога редактирования внешнего вида пера*. Данная кнопка индицирует текущий цвет линий пера;
- Название пера. При клике мыши возможно редактирование названия;
- Источник пера - переменная MasterSCADA;
- Текущие границы вертикального масштаба. При клике мыши возможно редактирование границ. Рядом со значениями границ размещены кнопки автомасштаба и масштаба шкалы, дублирующие кнопки на вспомогательной вертикальной панели окна графиков;
- Размерность пера;
- Значение пера в срезе вертикального курсора;
- Последнее значение пера;
- Кнопки управления *Видимостью уставок пера* - заданных границ переменной MasterSCADA. Если границы переменной не заданы на закладке переменной *Контроль значений* или для шкалы переменной, то кнопка соответствующей уставки невидима. Нажатие кнопки включает/выключает видимость заданной уставки. При включенной уставке – ее значение отображается на кнопке;
- Чекбокс управления видимости пера в минилегенде.

Для списка перьев может быть выбрано три типа группировки:

- По типу источника (аналоговые, дискретные, перечислимые, другие);
- По системной шкале пера;
- Группировка отключена.

Для столбцов панели может быть задана сортировка по возрастанию или убыванию. Сортировка переключается кликом мыши по заголовку панели. Размеры столбцов заголовка легенды можно изменять. Также можно менять столбцы местами. На панель перьев может быть добавлена переменная MasterSCADA путем перетаскивания ее из дерева объектов. Если переменная перетаскивается на пустое место – она добавляется как новое перо. Если перетаскивается на существующее перо – то происходит перепривязка источника пера. При клике мыши на строку с пером – выбранное перо становится *Активным пером*. При наведе-

нии мыши на строку с пером – оно подсвечивается в *Акне графиков*. Пользовательский интерфейс таблицы панели перьев соответствует общему интерфейсу таблиц.

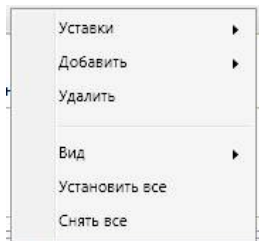
Панель разметки отображает список всех графических объектов разметки тренда. К разметке относятся все объекты, не являющиеся графиками: оси, координатные сетки, закладки, уставки, бордюры и другие.

Тулбар панели разметки содержит следующие кнопки, сверху вниз:

- *Переключение вида* - простой/детальный;
- *Группировка* - группировка по типу/группировка отключена. Переключение группировки осуществляется последовательными кликами мыши по кнопке. Включенное состояние группировки индицируется утопленной кнопкой.

Контекстное меню панели разметки содержит следующие команды:

- *Уставки* - меню включения/выключения уставок;
- *Добавить* - меню добавления новых объектов разметки;
- *Удалить* - удаляет выделенные объекты разметки;
- *Вид* - простой/детальный. Устанавливает простой или детальный вид;
- *Показать все* - делает все объекты разметки видимыми;
- *Скрыть все* - делает все объекты разметки невидимыми.



Вид панели разметки может быть простой или детальный. Простой вид отображает следующую информацию по каждому перу:

- Чекбокс видимости объекта;
- Название объекта.



Детальный вид отображает следующую информацию по каждому элементу разметки:

- Чекбокс видимости элемента;
- Кнопка вызова *Диалога редактирования внешнего вида*. Данная кнопка индицирует текущий цвет линий объекта;
- Название объекта. При клике мыши возможно редактирование названия.

Для списка объектов разметки может быть выбрано два типа группировки:

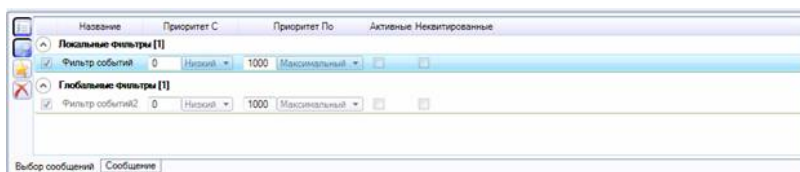
- По типу объекта;
- Группировка отключена.



Для столбцов панели может быть задана сортировка по возрастанию или убыванию. Сортировка переключается кликом мыши по заголовку панели. Размеры столбцов заголовка легенды можно изменять. Также можно менять столбцы местами.

При клике мыши на строку с объектом - он становится выбранным. При наведении мыши – подсвечивается в окне графиков. Пользовательский интерфейс таблицы панели разметки соответствует общему интерфейсу таблиц.

Панель выбора сообщений содержит список всех фильтров, относящихся к данному журналу.

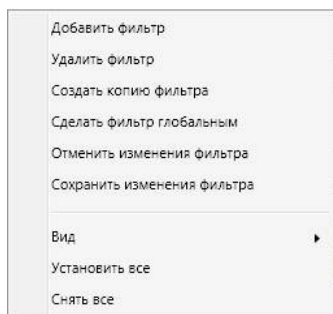


Тулбар панели выбора сообщений содержит следующие кнопки, сверху вниз:

- Переключение вида - простой/детальный;
- Группировка - вкл/выкл;
- Добавить фильтр;
- Удалить фильтр.

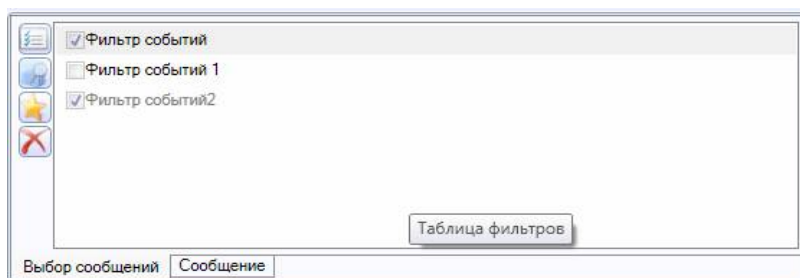
Контекстное меню панели выбора сообщений содержит следующие команды:

- *Добавить фильтр* - добавляет локальный фильтр;
- *Удалить фильтр* - удаляет фильтр;
- *Сделать копию фильтра* - создает локальную копию;
- *Сделать фильтр глобальным* - делает фильтр глобальным;
- *Отменить изменения* - отменяет изменения глобального фильтра;
- *Сохранить изменения* - сохраняет изменения глобального фильтра.



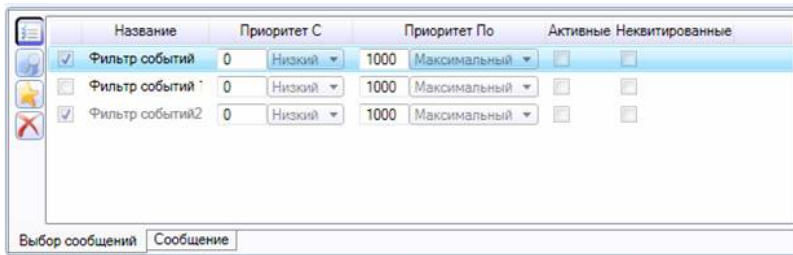
Вид панели может быть простой или детальный. Вид можно настроить из контекстного меню панели или из тулбара панели. Простой вид отображает следующую информацию по каждому фильтру:

- Чекбокс задействования фильтра;
- Название фильтра. По клику мыши можно переименовать.



Детальный вид отображает следующую информацию по каждому фильтру:

- Чекбокс задействования фильтра;
- Название фильтра. По клику мыши можно переименовать;
- Диапазон приоритетов сообщений;
- Флаг активных сообщений;
- Флаг неактивированных сообщений.



Для списка элементов разметки может быть включена или отключена группировка по типу фильтра:

- Локальные фильтры;
- Глобальные фильтры;
- Измененные фильтры.

При начале редактирования глобального фильтра - он перемещается в группу *Измененные*, это означает, что он переопределен только для данного журнала. То есть создается локальная копия глобального фильтра, которая скрывает глобальный фильтр. Что применить изменения ко всем журналам (применить изменения к исходному глобальному фильтру) - следует выполнить команду контекстного меню *Сохранить изменения*. Также можно *Отменить изменения*, при этом все изменения глобального фильтра будут отменены. Пользовательский интерфейс таблицы панели выбора сообщений соответствует общему интерфейсу таблиц.

Панель расширенной информации о сообщении содержит детальную информацию о выбранном сообщении.



Информация разделена на два столбца. Левый столбец содержит название свойства сообщения. Правый - значение свойства. Видимость свойств можно управлять из контекстного меню, отключая вывод ненужной информации.



Внизу панели выбора сообщений находится текстовое поле для ввода комментария, а также кнопка квитиования сообщения с заданным комментарием.

Панель горизонтального навигатора является инструментом для задания временных границ с помощью мыши. Влияет на все панели тренда, оперирующие понятием времени: *Окно графиков*, *Таблицу значений*, *Таблицу сообщений*, *Панель даты-времени*.



Навигатор состоит из девяти частей, слева направо:

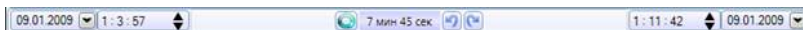
- Кнопка перехода к самому первому событию/значению за всю историю;
- Кнопка-повтор сдвига временного окна к более ранним данным на 10%;
- Кнопка-повтор сдвига временного окна к более ранним данным на 100%;
- Кнопка-слайдер изменения левой границы окна;
- Ползунок-слайдер сдвига временного окна;
- Кнопка-слайдер изменения правой границы окна;
- Кнопка-повтор сдвига временного окна к более поздним данным на 100%;
- Кнопка-повтор сдвига временного окна к более поздним данным на 10%;
- Кнопка перехода к самому последнему событию/значению за всю историю;

Панель вертикального навигатора является инструментом для задания вертикальных масштабов графиков с помощью мыши. Влияет на *Окно графиков*.

Навигатор состоит из девяти частей, сверху вниз:

- Кнопка перехода к самому верхнему значению. Верхнее значение равно или максимуму шкал всех графиков или максимальному видимому значению. В зависимости от того, что больше;
- Кнопка-повтор сдвига к верху шкал на 10%;
- Кнопка-повтор сдвига к верху шкал на 100%;
- Кнопка-слайдер изменения верхней границы видимого окна для всех графиков;
- Ползунок-слайдер сдвига видимого окна по вертикали;
- Кнопка-слайдер изменения нижней границы видимого окна для всех графиков;
- Кнопка-повтор сдвига к низу шкал на 100%;
- Кнопка-повтор сдвига к низу шкал на 10%;
- Кнопка перехода к самому нижнему значению. Нижнее значение равно или минимуму шкал всех графиков или минимальному видимому значению. В зависимости от того, что меньше.

Панель даты-времени является инструментом для задания временных границ.



Установленный интервал времени влияет на все панели тренда, оперирующие понятием времени: *Окно графиков*, *Таблицу значений*, *Таблицу сообщений*, *Горизонтальный навигатор*.

Панель даты/времени содержит следующие элементы слева направо:

- Элемент *Календарь* для выбора даты левой границы;
- Элемент *Время* для выбора времени левой границы;
- Кнопка управления автопрокруткой . Переключает режим автопрокрутки;
- Элемент быстрого выбора интервалов времени;
- Кнопка загрузки сохраненного состояния тренда (состояния, сохраненного в режиме разработки) ;
- Кнопки возврата временного состояния режима исполнения, после загрузки сохраненного состояния тренда ;
- Элемент *Время* для выбора времени правой границы;
- Элемент *Календарь* для выбора даты правой границы.

На фоне всех элементов расположен индикатор фоновых операций с архивами данных или сообщений, выполнение которых задерживается по каким-то причинам. Синий цвет показывает выполнение запросов к серверу данных. Зеленый цвет показывает выполнение запросов к серверу сообщений. Желтый цвет показывает выполнение запросов и к серверу данных и к серверу сообщений.

Элемент *Календарь* реализует следующее поведение:

- Дату можно выбрать из всплывающего окна;
- Дату можно задать вручную, вводом с клавиатуры. При смене даты - время устанавливается в 00:00:00.

Элемент *Время* реализует следующее поведение:

- Часы/минуты/секунды можно задать вручную, вводом с клавиатуры. При этом более младшие единицы времени обнуляются;
- Установив фокус ввода на один из разрядов можно нажатием кнопки-повтора справа увеличивать/уменьшать время на величину выбранного разряда. При переполнении разряда увеличивается более старший разряд или уменьшается младший, в зависимости от направления переполнения.

Элемент задания пользовательских периодов времени реализует следующее поведение во всплывающем окне:

- Установка выбранного временного интервала;
- Редактирование временного интервала в новом всплывающем окне. Редактирование интервала, базовой единицы времени, а также возможность полного удаления;
- По выбору пункта *Новый* добавление нового пользовательского интервала;
- По выбору пункта *Все данные* установка временного интервала для отображения всех значений/событий за всю историю;
- По выбору пункта *Сортировать* сортировка пользовательских интервалов по возрастанию длительности интервала.

При закрытом всплывающем окне - элемент задания пользовательских периодов времени отображает текущий временной интервал. Рядом с элементами выбора левой и правой границ – находятся чекбоксы фиксации значений границ. При установке фиксации – граница при любых действиях остается неизменной, чтобы бывает полезно в сценарии *Сжатия/растяжения* графиков.

Таблица сообщений служит для отображения списка сообщений в табличном виде.

Начало	Конец	Сообщение	Описание	Комментарий	Источник
2009.09.01 14:03:56		Наступило Событие 1	Событие 1		Событие 1
2009.09.01 14:03:41		Наступило Событие 2	Событие 2		Событие 2
2009.09.01 14:03:26	2009.09.01 14:03:40	Наступило Событие 1	Событие 1		Событие 1
2009.09.01 14:03:12		Максимальное аварийное	Значение		Значение
2009.09.01 14:03:11		Наступило Событие 3	Событие 3		Событие 3
2009.09.01 14:03:11	2009.09.01 14:03:12	Минимальное предупреждающее	Значение		Значение
2009.09.01 14:02:56	2009.09.01 14:03:10	Наступило Событие 1	Событие 1		Событие 1
2009.09.01 14:02:50	2009.09.01 14:02:51	Максимальное предупреждающее	Значение		Значение

Неквитированные сообщения (сообщения, прочтение которых оператор не квитировал, то есть не подтвердил) отображаются курсивом, **Активные** (актуальные) сообщения (сообщения, события которых еще не завершены) отображаются жирным шрифтом. Внешний вид сообщения (цвет фона и текста сообщения) задается при определении категории на закладке *Сообщения* узла *Система* дерева системы MasterSCADA.

Категория	Приоритет
Авария	Высокий
Предупреждение	Средний
Отсутствие значения	Средний
Информация	Низкий
Системные ошибки	Средний
Исполнительные механизмы	Обычный

Канал
<input checked="" type="checkbox"/> Сообщение
<input checked="" type="checkbox"/> Строка статуса
<input checked="" type="checkbox"/> Журнал
<input type="checkbox"/> Принтер
<input checked="" type="checkbox"/> Архив
<input type="checkbox"/> E-mail
<input type="checkbox"/> SMS

Звук... [Dropdown]

Добавить Удалить Звук... [Dropdown] Текст Фон

В таблице сообщений отображается следующая информация:

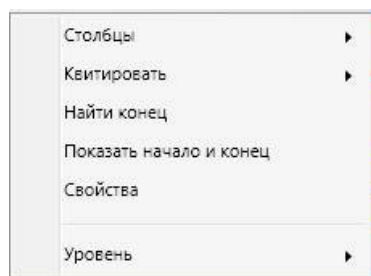
- Признак пометки сообщений. Служит для временной пометки сообщений с целью дальнейшего анализа;
- Кнопка квитирования сообщения. По нажатию кнопки соответствующее сообщение помечается как квитированное.

Остальные столбцы являются опциональными. Их видимость задается из контекстного меню таблицы сообщений. Видимость может быть включена для следующих столбцов:

- *Начало* - отображает время формирования сообщения;
- *Конец* - отображает время закрытия сообщения;
- *Сообщение* - отображает текст сообщения;
- *Описание* - отображает описание сообщения. Задается на закладке *Общие* у переменной MasterSCADA;
- *Комментарий* - отображает комментарий. Задается на закладке *Общие* у переменной MasterSCADA;
- *Источник* - отображает источник, сформировавший сообщение;
- *Изменения*;
- *Новое состояние* - отображает состояние сообщения;
- *Объект* - объект, содержащий источник сообщения;
- *Последнее квитирование* - время последнего квитирования;
- *Оператор* - оператор, квитировавший сообщение;
- *Тип сообщения* - отображает тип сообщения;
- *Тип источника* - отображает тип источника;
- *Категория* - отображает категорию сообщения;
- *Приоритет* - отображает приоритет сообщения;
- *Условие* - отображает условие формирования сообщения;
- *Подусловие* - отображает подусловие отображения сообщения;
- *Качество* - отображает качество сообщения;
- *Квитирование возможно* - отображает возможность квитирования сообщения;
- *Идентификатор* - уникальный идентификатор сообщения;
- *Примечание* - примечание, введенное оператором во время квитирования.

Контекстное меню таблицы сообщений содержит следующие команды:

- *Столбцы* - управление видимостью столбцов;
- *Квитировать* - возможность квитирования текущего сообщения, выделенных сообщений или всех сообщений с полем ввода комментария;
- *Навигация* - возможность установки масштаба времени таким образом, чтобы правая граница времени соответствовала концу текущего сообщения или чтобы левая граница сообщения – началу, или и то и другое;
- *Показать источник* - открывает новый тренд с источником сообщения или с источником и связями. Также возможно открытие тренда в режиме таблицы.



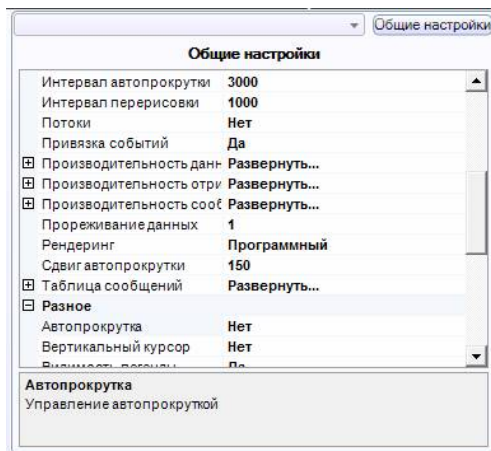
Квитирование выделенных сообщений с примечанием по умолчанию также можно сделать при помощи *Ctrl-Enter*. Пользовательский интерфейс таблицы сообщений соответствует общему интерфейсу таблиц. Для таблицы сообщений рекомендуется применять множественную сортировку по двум столбцам. Например, первичная сортировка по убыванию столбца *Конец*, а вторичная сортировка по убыванию столбца *Начало*. Это необходимо, так как времени конца у сообщения может не быть, а следовательно сортировка по столбцу *Конец* не определена и позиция сообщения в таблице строго недетерминирована.

Для отображения данных в табличном виде служит **Таблица значений**. Таблица значений отображает данные графиков, видимость которых включена в легенде. Данные отображаются в виде таблицы: в столбцах - значения, в строках – время. Для таблицы значений возможно задействование интерполяции данных, для этого следует нажать на кнопку с фиксацией *Режим интерполяции*, после чего в таблице будут отображаться интерполированные данные с заданным шагом интерполяции.

Режим интерполяции		Секунда		x 1
Дата/время	Значение °C	Значение		
09.01.2009 10:57:05.767	30,00	77,03		
09.01.2009 10:57:06.767	30,01	76,94		
09.01.2009 10:57:07.767	30,02	76,84		
09.01.2009 10:57:08.767	30,04	76,75		
09.01.2009 10:57:09.767	30,07	76,65		
09.01.2009 10:57:10.767	30,10	76,55		
09.01.2009 10:57:11.767	30,14	76,46		
09.01.2009 10:57:12.767	30,19	76,36		
09.01.2009 10:57:13.767	30,24	76,26		
09.01.2009 10:57:14.767	30,30	76,16		
09.01.2009 10:57:15.767	30,37	76,05		
09.01.2009 10:57:16.767	30,44	75,95		

Шаг интерполяции задается при помощи базового периода интерполяции, умноженного на заданный коэффициент.

Панель настроек служит для отображения настроек текущего выбранного объекта.

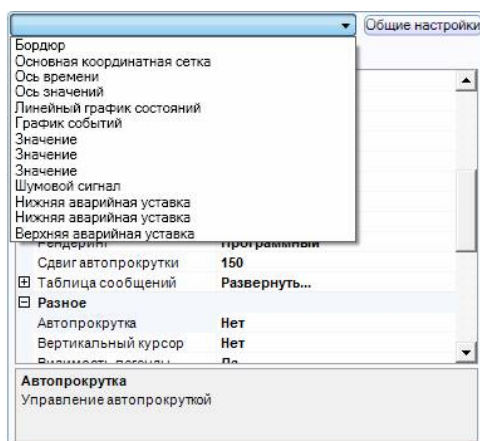


Панель настроек отображается в режиме настроек и может отображать настройки:

- Общие настройки тренда;
- Выбранного графического объекта;
- Выбранного фильтра сообщений.

Выбор общих настроек тренда для изменения производится нажатием на кнопку *Общие настройки* в панели настроек. Выбор графического объекта для изменения его настроек может производиться следующими способами:

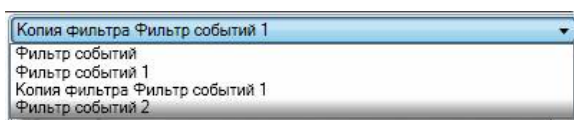
- Кликом мыши в окне графиков по объекту;
- Выбором объекта в панели перьев или панели разметки;
- Выбором объекта из выпадающего списка в панели настроек (только для режима тренда).



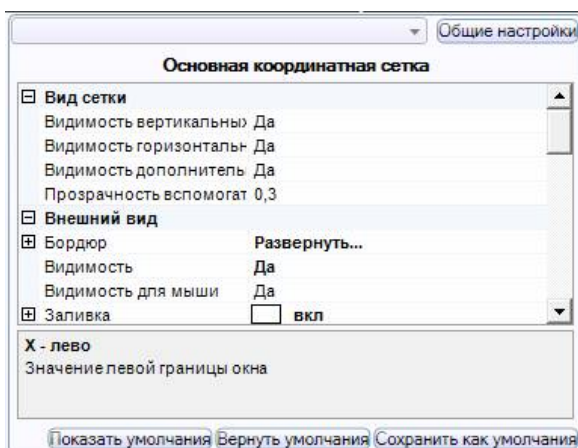
Выбор фильтра сообщений для редактирования настроек производится:

- Кликом мыши по фильтру в панели выбора сообщений;

- Выбором объекта из выпадающего списка в панели настроек (только для режима журнала).



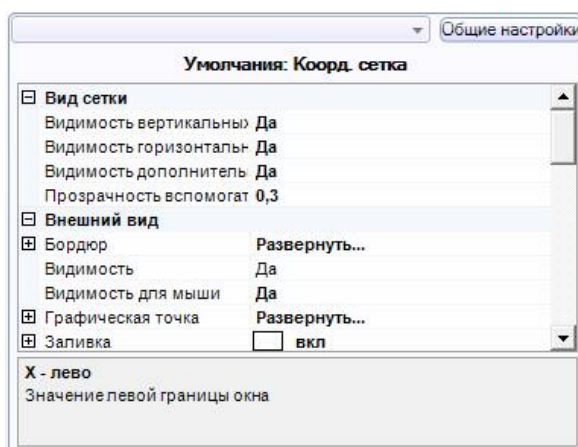
Выбор нескольких объектов осуществляется при зажатой кнопке *Ctrl*. В панели настроек существует возможность работать с умолчаниями настроек (умолчания настроек заданы для всех графических объектов). Редактирование умолчаний позволяет установить значение настройки одновременно для всех объектов данного типа, если это значение не было переопределено в объекте. На рисунке ниже приведены настройки объекта, имеющего умолчания.



Внизу панели настроек для такого объекта доступны кнопки:

- *Показать умолчания* - отображение настроек по умолчанию;
- *Вернуть умолчания* - применение настроек по умолчанию к данному объекту. При этом все переопределенные настройки сбрасываются к умолчаниям;
- *Сохранить как умолчания* - сохраняет настройки текущего объекта как умолчания.

На рисунке ниже приведены настройки умолчаний объекта (при нажатии на кнопку *Показать умолчания*).



Настройки, которые были переопределены – выделяются жирным шрифтом. Настройки, которые наследуются от умолчаний, отображаются обычным шрифтом. При изменении умолчанию такие настройки будут также изменены.

Тулбар тренда служит для размещения элементов управления наиболее востребованными функциями.



Тулбар включает в себя четыре панели:

- Панель экспорта;
- Функциональная панель;

- Панель управления видимостью;
- Панель управления слоями данных.






Панель экспорта содержит три выпадающих меню:

- Скриншот;
- Сохранение в файл;
- Печать.

В зависимости от видимости различных панелей тренда эти меню могут содержать следующие команды:

- Скриншот тренда (сохранение снимка экрана всех видимых панелей тренда в один файл);
- Скриншот графиков (сохранение снимка экрана окна графиков);
- Скриншот значений (сохранение снимка экрана таблицы значений);
- Скриншот сообщений (сохранение снимка экрана таблицы сообщений);
- Сохранение значений (экспорт данных из таблицы значений в файл, в формате *csv*);
- Сохранение сообщений: экспорт сообщений из таблицы сообщений в файл, в формате *html*;
- Печать тренда;
- Печать графиков;
- Печать сообщений;
- Печать значений.

Наличие кнопок на **функциональной панели** может меняться. В зависимости от уровня сложности и режима открытия могут быть видны следующие кнопки:

- *Загрузка настроек типа журнала* . Происходит полное восстановление настроек, определенных в типе журнала;
- *Обновление всех видимых данных и сообщений* . Происходит полное обновление всего содержимого тренда;
- *Импорт тренда из сохраненного состояния другого тренда* . Открывается диалог выбора другого тренда. Настройки выбранного тренда загружаются в текущий;
- *Вызов справки* . Отображение справки по тренду;
- *Выбор перьев* . Открывается диалог выбора перьев.

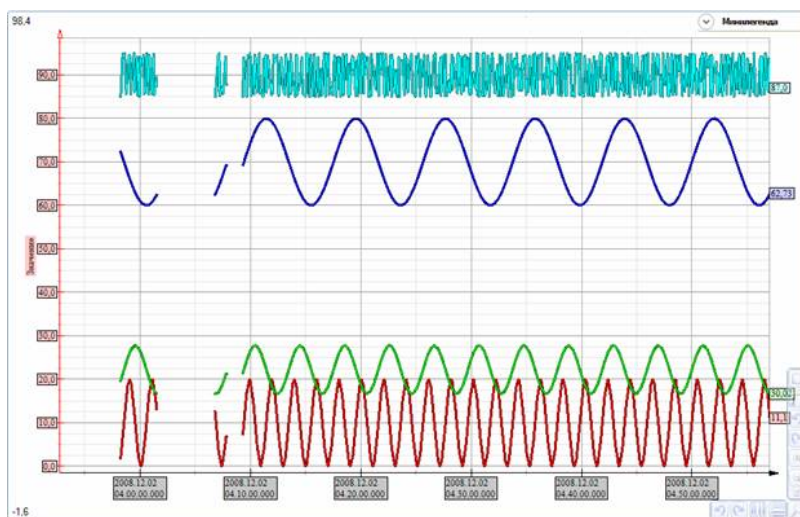
Панель управления видимостью служит для включения и отключения видимости других панелей на экране, панель содержит следующие кнопки с фиксацией:

- *Графики* - управление видимостью окна графиков;
- *Значения* - управление видимостью таблицы значений;
- *Сообщения* - управление видимостью таблицы сообщений;
- *Легенда* - управление видимостью панели легенды;
- *Настройки* - управление видимостью панели настроек.

Панель управления слоями данных позволяет вручную управлять отображаемым слоем данных. Содержит кнопку с фиксацией для ручного выбора слоя данных, а также комбобокс со списком слоев данных.

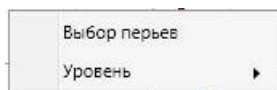


Окно графиков – прямоугольная область, в которой расположены все графические объекты.



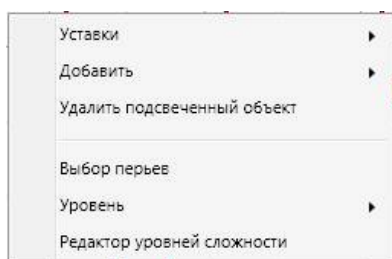
Основное контекстное меню окна графиков содержит следующие команды:

- *Выбор перьев* - открывает диалог выбора перьев;
- *Уровень* - содержит список шаблонных и пользовательских уровней сложности, на которые у оператора есть права.



В режиме настроек появляются следующие подменю:

- *Уставки*. Содержит команды управления видимостью уставок. Если у активного графика нет уставки - она добавляется. Если есть уставка - она удаляется. Уставка идентифицируется по позиции (значению). Значения уставок из этого списка полностью соответствует значениям границ переменной MasterSCADA, являющейся источником активного графика;
- *Добавить*. Содержит список графических объектов и обработок, которые можно добавить в *Окно графиков*;
- *Удалить подсвеченный объект*. Удаляет подсвеченный объект, если это можно сделать;
- *Уровень*: содержит список шаблонных и пользовательских уровней сложности, на которые у оператора есть права;
- *Редактор уровней сложности*. Открывает диалоговое окно *Редактора уровней*;
- *Показать связи*. Показывает связи выделенных перьев в отдельном тренде;
- *Воспроизведение*. Включает *Режим воспроизведения истории*.



Работу с окном графиков можно производить посредством мыши и клавиатуры. При перемещении мыши происходит подсветка графического объекта под курсором мыши.

Левая кнопка мыши:

- Клик производит выделение графического объекта (графика или разметки);
- Нажатие кнопки и последующее перемещение мыши производит перемещение выделенного графического объекта. Если выделенного объекта нет, то происходит перемещение окна графиков.

Правая копка мыши:

- Клик вызывает контекстное меню;

- Нажатая правая кнопка и последующее перемещение мыши в режиме настроек производит вертикальное масштабирование выделенного графика относительно позиции нажатия. Если выделенного графика нет, то происходит вертикальное масштабирование всего окна графиков.

Колесо мыши:

- Прокрутка колеса осуществляет горизонтальное масштабирование окна графиков;
- Прокрутка колеса с нажатой правой кнопкой в режиме настроек осуществляет вертикальное масштабирование выделенного графика. Если выделенного графика нет, то происходит вертикальное масштабирование всех графиков.

Интерфейс клавиатуры:

- *F1* - отображение справки для объекта под курсором мыши;
- *Ctrl + Alt + F* - отображение отладочной информации;
- *Ctrl + Пробел* - вкл /выкл автопрокрутку в режиме исполнения;
- *Ctrl + N* - вкл/выкл режим настроек;
- *Ctrl + T* - вкл/выкл таблицу значений;
- *Ctrl + M* - вкл/выкл таблицу сообщений;
- *Ctrl + G* - вкл/выкл окно графиков;
- *Ctrl + L* - вкл/выкл легенду;
- *Ctrl + V* - вкл/выкл вертикальный курсор;
- *Ctrl + U* - вкл/выкл горизонтальный курсор;
- *Ctrl + +*: увеличить масштаб относительно центра на 10%;
- *Ctrl + -* - уменьшить масштаб относительно центра на 10%;
- *Ctrl + Стрелка вправо* - сдвинуть выделенный объект вправо;
- *Shift + Стрелка вправо* - сдвинуть выделенный объект вправо с шагом 1/100 экрана;
- *Ctrl + Shift + Стрелка вправо* - сдвинуть выделенный объект вправо с шагом 1/10 экрана;
- *Ctrl + Стрелка влево* - сдвинуть выделенный объект влево;
- *Shift + Стрелка влево* - сдвинуть выделенный объект влево с шагом 1/100 экрана;
- *Ctrl + Shift + Стрелка влево* - сдвинуть выделенный объект влево с шагом 1/10 экрана;
- *Ctrl + Стрелка вверх* - сдвинуть выделенный объект вверх;
- *Ctrl + Стрелка вниз* - сдвинуть выделенный объект вниз;
- *Ctrl + Enter* – квитирование выделенных сообщений при наличии фокуса ввода у таблицы сообщений;
- *Ctrl + левая кнопка мыши* – выделение нескольких графических объектов одновременно.


Для работы горячих клавиш окно графиков должно иметь фокус ввода. Фокус ввода можно установить кликом мыши в окне.


Режимы окна графиков - это состояния, в которых обеспечивается дополнительная функциональность или изменяется интерфейс пользовательского ввода. Несколько режимов может быть включено одновременно.

Графические курсоры предназначены для получения срезов графиков в заданный момент времени или по заданному значению.

В тренде существует три режима графического курсора:

- Режим вертикального курсора;
- Режим горизонтального курсора;
- Режим перекрестия.

В режиме *вертикального курсора* становится видимым вертикальный курсор. *Вертикальный курсор* - это вертикальная линия в окне графиков, которая следует за курсором мыши. Вертикальный курсор отображает значения (временные срезы) графиков и других объектов, с которыми он пересекается. Двойной клик левой кнопки мыши устанавливает закладку в текущей позиции. Режим вертикального курсора переключается с *панели горизонтального масштаба* кнопкой управления *вертикальным курсором*  или в *общих настройках тренда*. Положение курсора по X отображается внизу линии курсора. В режиме горизонтального курсора становится видимым горизонтальный курсор.


Горизонтальный курсор - это горизонтальная линия в окне графиков, которая следует за курсором мыши. Горизонтальный курсор отображает время (срезы значений) графиков и других объектов, с которыми он пересекается. Двойной щелчок левой кнопки мыши устанавливает уставку для активного графика в текущей позиции. Режим горизонтального курсора переключается с панели горизонтального масштаба кнопкой управления горизонтальным курсором  или в общих настройках тренда. Положение курсора в шкале активного графика по Y отображается слева линии курсора.


Режим перекрестия объединяет режимы вертикального и горизонтального курсоров. Этот режим задействован когда включены оба режима. В режиме перекрестия по центру перекрестия отображается текущее положение перекрестия по X и Y в окне графиков.

Режим редактирования задатчика предназначен для редактирования опорных точек задающего графика. В режиме редактирования задатчика становятся видны задающие точки активного графика (табличные значения), а также изменяется интерфейс мыши.

Изменения интерфейса левой кнопки мыши:

- Двойной клик по задающей точке удаляет ее с выделенного графика;
- Двойной клик по пустому месту добавляет задающую точку на выделенный график;
- Нажатие кнопки на задающей точке с последующим движением мыши (драг) сдвигает ее, изменяя вид выделенного графика.

Режим редактирования задатчика переключается с панели горизонтального масштаба кнопкой редактирования задатчика  или в общих настройках тренда. Кнопка редактирования задатчика видна только в продвинутом уровне сложности и только в режиме настроек. Режим редактирования задатчика включается автоматически при добавлении табличного задающего графика.


Режим лупы предназначен для группового масштабирования графиков. В режиме лупы изменяется интерфейс мыши. Режим лупы переключается кнопкой лупы .

Изменение интерфейса левой кнопка мыши:

- Клик увеличивает масштаб окна относительно центра на 10%;
- Нажатие кнопки с последующим движением мыши (драг) отображает подсвеченный прямоугольник части окна, к которой будет произведено масштабирование после отпущения кнопки.

Изменение интерфейса правой кнопка мыши:

- Клик уменьшает масштаб окна относительно центра на 10%.

Режим автопрокрутки предназначен для того, чтобы в режиме исполнения отображать на экране всегда самые актуальные данные. При включенном режиме автопрокрутки текущее время и соответственно данные всегда находятся в области видимости. В режиме автопрокрутки происходит отключение некоторых визуальных элементов, например элементов задания даты/времени и кнопок навигации по маркерам. То, какие именно элементы отключаются - зависит от текущего уровня сложности тренда. Режим автопрокрутки переключается кнопкой автопрокрутки  на *Панели даты-времени*. Эта кнопка видна только в режиме исполнения.

Режим воспроизведения истории предназначен для воспроизведения архивной истории с заданной скоростью. Режим воспроизведения включается из контекстного меню окна графиков. Там же, в контекстном меню можно управлять скоростью воспроизведения, а также интервалом сдвига.

Режим настроек предназначен для настройки содержимого тренда. В режиме настроек можно изменять расположение любых графических объектов разметки, в том числе по вертикали. Можно также добавлять и удалять графические объекты разметки. В режиме настроек видна панель настроек. Переход в режим настроек осуществляется при помощи кнопки *Настройки* в тулбаре тренда.

Панель горизонтального масштаба содержит следующие кнопки слева направо:

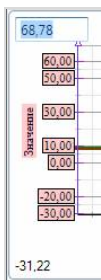
- Кнопка перехода к предыдущей закладке осуществляет переход к предыдущей закладке, ее выделение и центрирование;
- Кнопка перехода к следующей закладке осуществляет переход к следующей закладке, ее выделение и центрирование;

- Кнопка отмены изменения горизонтальных границ отменяет сдвиг или масштабирование по горизонтали;
- Кнопка возврата изменения горизонтальных границ возвращает отмененный сдвиг или масштабирование по горизонтали;
- Кнопка вертикального курсора. Кнопка с фиксацией включает/выключает режим вертикального курсора;
- Кнопка горизонтального курсора. Кнопка с фиксацией включает/выключает режим горизонтального курсора;
- Кнопка редактирования задатчика включает/выключает режим редактирования задатчика.

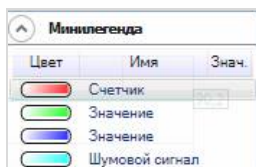
Панель вертикального масштаба содержит следующие кнопки сверху вниз:

- Кнопка разового подбора автоматического масштаба. Устанавливает масштаб выделенного графика так, чтобы он целиком был виден на экране. Если ни один график не выделен, то подбор осуществляется для всех графиков;
- Кнопка масштаба шкалы. Устанавливает масштаб выделенного графика в соответствии с пределами шкалы. Если ни один график не выделен, то масштабирование осуществляется для всех графиков;
- Кнопка отмены изменения вертикальных границ отменяет сдвиг или масштабирование по вертикали;
- Кнопка возврата изменения вертикальных границ возвращает отмененный сдвиг или масштабирование по вертикали;
- Кнопка увеличения масштаба увеличивает масштаб выбранного графика на 10% относительно центра экрана. Если ни один график не выделен, то масштабирование осуществляется для всего окна графиков;
- Кнопка уменьшения масштаба. Уменьшает масштаб выбранного графика на 10% относительно центра экрана. Если ни один график не выделен, то масштабирование осуществляется для всего окна графиков;
- Кнопка сдвига вверх. Сдвигает выбранный объект вверх в окне графиков. Если ни один объект не выбран – сдвигает окно графиков целиком;
- Кнопка сдвига вниз. Сдвигает выбранный объект вниз в окне графиков. Если ни один объект не выбран – сдвигает окно графиков целиком;
- Кнопка выбора пользовательского вертикального масштаба. При клике на кнопке появляется меню с предустановленными вертикальными интервалами, заданными в процентах. Любой интервал можно отредактировать, удалить или добавить новый. Выбранный масштаб применяется к выбранному перу, или ко всем перьям, если ни одно не выбрано.

Элементы управления вертикальными границами отображают вертикальные границы *Активного графика*, а также позволяют изменить их. Одинарный клик по элементу переводит его в режим редактирования. Двойной клик по элементу выделяет весь текст.



Минилегенда отображает список перьев. В отличие от панели перьев этот список максимально упрощен. Минилегенда отображает цвет пера, его название и последнее значение. На перо в минилегенде можно перетащить (*DRAG&DROP*) элемент дерева объектов MasterSCADA, при этом произойдет перепривязка источника пера. Минилегенда может находиться в развернутом состоянии как на рисунке ниже.



Или в свернутом состоянии, чтобы не мешать просмотру графиков.



Свертывание/развертывание минилегенды осуществляется кликом мыши по шапке легенды. По умолчанию в минилегенде отображаются все перья, которые отображаются в панели перьев, но отображение отдельных перьев в минилегенде (например перьев-уставок) можно запретить из панели перьев. Для столбцов панели может быть задана сортировка по возрастанию или убыванию. Сортировка переключается кликом мыши по заголовку панели. Размеры столбцов заголовка легенды можно изменять. Также можно менять столбцы местами.

На панель перьев может быть добавлена переменная MasterSCADA путем перетаскивания ее из дерева объектов. Если переменная перетаскивается на пустое место – она добавляется как новое перо. Если перетаскивается на существующее перо – то происходит перепривязка источника пера. При клике мыши на строку с пером – выбранное перо становится активным пером. При наведении мыши на строку с пером – оно подсвечивается в окне графиков. Пользовательский интерфейс таблицы минилегенды соответствует общему интерфейсу таблиц.

Графические объекты - логически законченные и независимые элементы в окне графиков. При создании нового тренда по умолчанию создаются следующие графические объекты:

- Окно графиков;
- Бордюр;
- Основная координатная сетка;
- Ось значений;
- Ось времени;
- Линейный график состояний;
- График событий.

При добавлении переменной MasterSCADA на тренд создается новый график, источником которого служит переменная MasterSCADA. Все остальные графические объекты, например уставки, обработки графиков, задающие графики - можно добавить из контекстного меню окна графиков. Все настройки графических объектов группированы по категориям. Общие свойства всех графических объектов предоставлены далее.

Категория Окно X. Содержит свойства, влияющие на горизонтальное измерение:

- *Форматер X.* Форматер значений X. Правила задания и описание форматера описаны ниже;
- *X-лево.* Левая граница окна в единицах, заданных форматтером;
- *X-право.* Правая граница окна в единицах, заданных форматтером;
- *Шкала X.* Шкала значений X может быть линейная или логарифмическая.

Категория Окно Y. Содержит свойства, влияющие на вертикальное измерение:

- *Y-верх.* Верхняя граница окна в единицах, заданных форматтером;
- *Y-низ.* Нижняя граница окна в единицах, заданных форматтером;
- *Форматер Y.* Форматер значений Y. Правила задания и описание форматера описаны ниже;
- *Шкала Y.* Шкала значений Y может быть линейная или логарифмическая.

Категория Внешний вид. Содержит свойства, влияющие на внешний вид:

- *Бордюр.* Задает цвет и стиль линий окантовки контура графического объекта;
- *Заливка.* Задает цвет заливки области, ограниченной контуром графического объекта;
- *Линия.* Задает цвет и стиль линий графического объекта;
- *Шрифт.* Задает стиль текстовых подписей графического объекта;
- *Графическая точка.* Задает стиль вспомогательных точек (маркеров) графического объекта;
- *Видимость.* Отвечает за видимость графического объекта.

Категория Связанные свойства. Содержит связи категорий двух объектов. Например если связать категорию *Окно Y* для двух графиков, то все их свойства, принадлежащие категории *Окно Y* окажутся идентичны и будут в дальнейшем поддерживать идентичность при изменении. То есть при изменении масштаба одного графика будет одновременно изменяться масштаб второго графика. Также если связать категорию *Внешний вид* для графика и вертикальной оси – то цвет оси будет совпадать с цветом графика. При создании тренда по умолчанию устанавливаются следующие связи графических объектов:

- *Окно X* всех объектов связано, то есть границы X едины для всех объектов. Это позволяет установить границы X (время) из свойств любого объекта: графика, оси, координатной сетки;
- *Окно Y*, *Разметка сетки X*, *Разметка сетки Y* координатной сетки связаны с активным графиком. Это означает, что масштаб и разметка узлов сетки берется от текущего активного графика. При смене активного графика будет отображаться именно его индивидуальная сетка. Если разорвать эти связи (установить в пустое значение), то координатная сетка будет одна для всех графиков;
- *Внешний вид* вертикальной оси связан с активным графиком. Это значит, что цвет оси будет меняться при смене активного графика.

Для любых кистей можно выбрать вариант заливки:

- Солидная заливка одним цветом;
- Градиентная заливка двух цветов;
- Заливка выбранной картинкой.

Координатная сетка - сетка из вертикальных и горизонтальных линий, служащая для облегчения восприятия информации. У координатной сетки может быть включено отображение мини-сетки – вспомогательных полупрозрачных линий между основными линиями, для которых не выводятся подписи осей. У каждого графика своя индивидуальная координатная сетка. По умолчанию тип сетки для всех новых графиков – автоматический. Настройки координатной сетки описаны ниже.

Категория *Вид сетки*:

- *Видимость вертикальных линий* управляет видимостью вертикальных линий сетки;
- *Видимость горизонтальных линий* управляет видимостью горизонтальных линий сетки;
- *Видимость дополнительной сетки линий* управляет видимостью вспомогательных линий сетки;
- *Прозрачность вспомогательных линий сетки* управляет прозрачностью вспомогательных линий сетки.

Категория *Разметка сетки X*:

- *Тип столбцов* определяет алгоритм расположения узлов вертикальных линий сетки. В зависимости от алгоритма появляются дополнительные настройки. Типы сетки описаны ниже;
- *Вспомогательные столбцы* определяют количество вспомогательных вертикальных линий сетки;
- *Минимум между линиями* задается в пикселах и определяет минимально допустимое расстояние между вертикальными линиями сетки, чтобы не допустить их скручивания. Если расстояние между двумя соседними линиями сетки меньше минимума – одна из линий не отображается.

Категория *Разметка сетки Y*:

- *Тип строк* определяет алгоритм расположения узлов горизонтальных линий сетки. В зависимости от алгоритма появляются дополнительные настройки. Типы сетки описаны ниже;
- *Вспомогательные столбцы* определяют количество вспомогательных горизонтальных линий сетки;
- *Минимум между линиями* задается в пикселах и определяет минимально допустимое расстояние между горизонтальными линиями сетки, чтобы не допустить их скучивания. Если расстояние между двумя соседними линиями сетки меньше минимума – одна из линий не отображается.

Типы сеток:

- *Автоматическая сетка*. Эта сетка отображает линии так, чтобы подписи значений на осях представляли собой округленные значения. Например: 10 20 30 40 50 60 или 0.1 0.2 0.3 0.4 0.5 0.6. У автоматической сетки нет никаких дополнительных настроек;
- *Статичная сетка*. Количество узлов сетки всегда фиксировано и не меняет положения на экране. Количество вертикальных узлов определяет свойство *столбцы*, количество горизонтальных узлов определяет свойство *строки*. Если например задано 5 строк, то координатная сетка делится по вертикали на 5 равных частей, в которых располагаются узлы⁴

- *Плавающая сетка*. То же самое, что и статичная сетка, только перемещающаяся по экрану при изменении масштаба. Начало этой сетки всегда привязано к определенному значению оси. Например начало сетки по горизонтали может быть привязано к 01.01.2000. Или по вертикали привязано к 100. Эта точка отсчета задается свойством *Точка отсчета*;
- *Кратная сетка*. Узлы этой сетки следуют с заданной кратностью, например каждый час или каждые 20 значений по оси Y. Эта кратность задается свойством *Кратность*;
- *Подогнанная сетка*. То же самое, что кратная сетка, только с точкой отсчета. Например если задать кратность 10 и точку отсчета 0, то узлы сетки всегда будут следующие: 0 10 20 30 40. Этот тип сетки используется для отображения дискретных значений. В этом случае точка отсчета равна 0, а кратность равна 1.

При настройке сеток следует учитывать, что настройка *Минимум между линиями* не позволит выводить слишком частую сетку и будет приводить к прореживанию.

Ось - это вертикальная или горизонтальная линия, отображающая значения связанного графика. Насечки и подписи оси располагаются в узлах координатной сетки графика. Ось унаследована от координатной сетки, а значит, содержит все ее настройки. Дополнительные настройки оси описаны ниже.

Категория Вид оси:

- *Вертикальное выравнивание названия*. Задает вертикальное выравнивание названия оси;
- *Вертикальное выравнивание насечек*. Задает вертикальное выравнивание насечек оси (место вывода подписей);
- *Вертикальное выравнивание подписей*. Задает вертикальное выравнивание подписей оси;
- *Видимость названия*. Определяет видимость названия оси;
- *Видимость стрелки*. Определяет видимость стрелки оси;
- *Видимость текста насечек*. Определяет видимость текста насечек (подписей);
- *Горизонтальное выравнивание названия*. Задает горизонтальное выравнивание названия оси;
- *Горизонтальное выравнивание насечек*. Задает горизонтальное выравнивание насечек оси (место вывода подписей);
- *Горизонтальное выравнивание подписей*. Задает горизонтальное выравнивание подписей оси;
- *Длина насечек*. Определяет длину насечек в пикселах;
- *Длина вспомогательных насечек*. Определяет длину вспомогательных насечек в пикселах;
- *Подпись оси*. Определяет подпись оси;
- *Тип оси*. Определяет тип оси: горизонтальная или вертикальная;
- *Тип стрелки*. Определяет тип стрелки оси.

Категория Позиция:

- *Горизонтальная привязка позиции* определяет горизонтальное выравнивание оси в окне графиков;
- *Вертикальная привязка позиции* определяет вертикальное выравнивание оси в окне графиков;
- *Смещение, %* определяет смещение оси относительно начала системы координат.

График - это упорядоченная во времени последовательность значений, с определенным законом аппроксимации геометрии между значениями. График унаследован от координатной сетки, а значит, содержит все ее настройки.

Дополнительные настройки графика:

- *Авторазмер*. Определяет должен ли график занимать всю видимую область автоматически или должен иметь жестко заданный пользовательский масштаб;
- *Дорисовка неизменных значений*. Определяет, следует ли дорисовывать неизменные значения графика. Неизменные значения определяется так: если за последние пять секунд не поступало новых данных, то график дорисовывается до текущего времени пунктирной линией;
- *Последнее значение*. Определяет следует ли отображать последнее значение графика

- *Сегменты*. Определяет тип сегментов графиков. *Линия* - соединяет соседние точки графика линией, применяется для аналоговых графиков. *Ступенька* - соединяет соседние точки графика ступенькой, применяется для дискретных графиков. *Безье2* и *Безье3* соединяют соседние точки графиков линиями *Безье*;
- - Способ заливки. Определяет способ заливки графика. Сама по себе эта настройка не включает заливку. Заливку следует включить в категории *Внешний вид*;
- *Таблица*. Вызывает редактор табличных значений;
- *Дополнительные настройки*. Отображает дополнительные настройки пера. *Источник* показывает связанный с пером источник. *Пользовательское имя* показывает название графика, заданное пользователем. *Размерность* отображает размерность источника. *Скрывать в мини-легенде* отображает необходимость сокрытия графика в мини-легенде. *Стиль имени* позволяет установить имя графика из источника в соответствии с принятой классификацией MasterSCADA. *Тип* - определяет тип графика.

Задающий график - это график, заранее определенный пользователем, в отличие от графика переменной MasterSCADA, значения которого заранее неизвестны и определяются технологическим процессом. Задающий график можно задать таблицей, формулой или тем и другим. При добавлении табличного задающего графика на тренд – устанавливается *Режим редактирования задатчика*. При добавлении задающего графика, определенного формулой – открывается *Диалог редактирования формулы графика*. Задающий график унаследован от графика и содержит его настройки. Дополнительные настройки задающего графика описаны ниже.

Категория *График*:

- *Формула* - открывается диалог *Редактирования формулы*;
- *Пользовательская таблица* - открывается диалог *Редактирования табличных значений*;
- *Зациклить* - возможность включить зацикливание, при этом табличное задание будет повторяться циклично в будущем.

Уставка - линия, связанная с графиком, которая показывает цветом переход графика через свою геометрию. Геометрия уставки может быть определена другим графиком. Если геометрия уставки не определена - она представляет собой горизонтальную линию. Настройка уставок производится в режиме настроек. В этом режиме доступно перемещение уставки мышью (изменение значения), а также настройка свойств уставки: цвета, внешнего вида, способа заливки, направления уставки, области действия и других свойств. Для настройки уставки как параметрической следует в ее свойствах задать *Объект геометрии* (в выпадающем списке произвести выбор среди уже добавленных перьев). Для смены привязки уставки к перу следует в ее свойствах задать *Объект контроля*.

Категория *Вид линии*:

- *Вертикальное выравнивание подписи*. Определяет вертикальное выравнивание подписи значения уставки;
- *Вертикальное выравнивание текста срезов*. Определяет вертикальное выравнивание подписей срезов;
- *Видимость позиции* определяет видимость подписи позиции уставки;
- *Горизонтальное выравнивание подписи*. Определяет горизонтальное выравнивание подписи значения уставки;
- *Горизонтальное выравнивание текста срезов*. Определяет горизонтальное выравнивание подписей срезов;
- *Текст срезов* определяет видимость подписей срезов.

Категория *Внешний вид уставки*:

- *Заливка*. Определяет тип заливки. Внутренняя заливка – это заливка между линией уставки и контуром графика. Внешняя заливка – это заливка между границей окна графиков и контуром графика;
- *Направление*. Определяет направление действия уставки. Верхняя уставка отображает превышение заданного значения. Нижняя уставка отображается при снижении заданного значения;
- *Область действия* определяет область действия уставки. Бесконечная область действия означает, что цвета соседних однонаправленных уставок будут смешиваться при наложении. Область действия *До следующей уставки* означает, что цвет уставки будет действовать только до следующей уставки, без наложения;

- *Положение*. Определяет положение уставки по вертикали в масштабе ее окна Y .

Категория *Связанные свойства*:

- *Объект контроля*. Определяет график, с которым связана уставка;
- *Объект геометрии*. Задаёт объект, который определяет геометрию уставки. Если это свойство не задано – уставка представляет собой горизонтальную линию;
- *Окно Y* . Определяет масштаб значений уставки. Значение этого свойства должно быть графиком, с которым связана уставка для того, чтобы вертикальные масштабы графика и уставки совпадали.

Закладка - это вертикальная линия, привязанная ко времени. Закладка показывает временные срезы графиков. Также закладки служат для облегчения навигации при просмотре архивной истории. Навигация между закладками осуществляется при помощи кнопок перехода к предыдущей и следующей закладке в панели горизонтального масштаба.

Категория *Вид линии*:

- *Вертикальное выравнивание подписи*. Определяет вертикальное выравнивание подписи значения уставки;
- *Вертикальное выравнивание текста срезов*. Определяет вертикальное выравнивание подписей срезов;
- *Видимость позиции* определяет видимость подписи позиции уставки;
- *Горизонтальное выравнивание подписи*. Определяет горизонтальное выравнивание подписи значения уставки;
- *Горизонтальное выравнивание текста срезов*. Определяет горизонтальное выравнивание подписей срезов;
- *Текст срезов* определяет видимость подписей срезов.

Категория *Положение X* :

- *Позиция* определяет текущее положение закладки.

Обработка графика – это другой график, данные которого получены из исходных данных трансформацией по некоторому алгоритму. Обработки делятся на следующие типы:

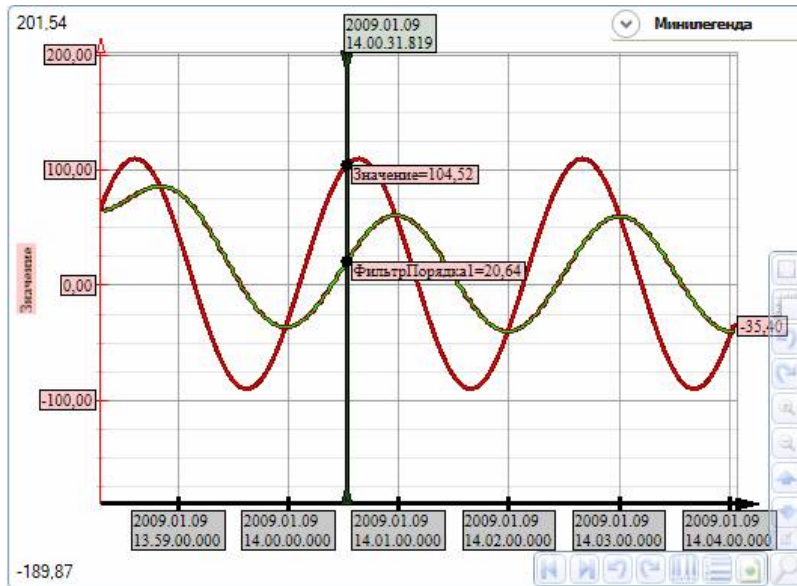
- *Математические* - содержат математические операции над графиков;
- *Фильтры* - задают фильтрацию данных графика;
- *Операции* - задают операции между данными двух графиков.

Добавление обработок, производится из контекстного меню окна графиков. Так как обработка является пером, то на нее может быть также наложена другая обработка, а также для нее применимы все операции с перьями в целом. *Математические обработки* – это обработки, исходными данными которых являются данные одного графика, результатом которых является математическое преобразование исходных данных. Настройки математической обработки:

- *Интервал*. Определяет кол-во значений, участвующих в групповой обработке. Например количество значений, по которым вычисляется среднее значение (интервал, за который происходит усреднение). Интервал - безразмерная величина, показывает количество исходных значений. Если интервал равен нулю, то он выбирается автоматически как $1/10$ количества значений в текущем окне просмотра. Если интервал больше количества значений (например 1000), обработка рассчитывается для всего окна и отображается в виде константы;
- *Действие*. Определяет тип трансформации значений;
- *Аргумент*. Определяет график, трансформация которого осуществляется.

Имеются следующие математические обработки: натуральный логарифм, десятичный логарифм, экспонента, степень двойки, степень десяти, производная, интеграл.

Фильтры – это обработки, исходными данными которых являются данные одного графика, результатом которых являются отфильтрованные исходные данные. Настройки фильтра аналогичны настройкам математических обработок. Имеются следующие фильтры: *среднее, экспоненциальное (скользящее) среднее, среднее квадратичное отклонение, дисперсия, фильтр первого порядка, минимальное, максимальное*. На рисунке ниже показана обработка *Фильтр первого порядка* для исходного синусоидального графика:



Операции - это обработки, исходными данными которых являются данные двух графиков, а результатом – график операции между этими данными.

Настройки операций:

- **Действие.** Определяет операцию между данными двух графиков;
- **Аргумент X.** Определяет график, являющийся первым аргументом операции;
- **Аргумент Y.** Определяет график, являющийся вторым аргументом операции.

Перечень операций: *сумма, разность, произведение, частное, остаток, среднее, зависимость, корреляция.*

- ❖ **Важно!** Операция *зависимость* представляет собой обычный график зависимости одного параметра от другого $Y=F(X)$. Если желательно отображение обработки и исходного графика в одинаковых масштабах - следует задать связь с исходным графиком по категории **Окно Y** в настройках.

Бордюр - это окантовка окна графиков. Предназначен для размещения дополнительных осей, подписей графических объектов, а также для размещения вспомогательных панелей. Дополнительные настройки бордюра:

- Ширина слева задает в пикселах отступ слева;
- Ширина сверху задает в пикселах отступ сверху;
- Ширина справа задает в пикселах отступ справа;
- Ширина снизу задает в пикселах отступ снизу.

Динамические звенья являются разновидностью задающих графиков. По сути это и есть задающие графики, для которых задана формула звена. Добавление звеньев, производится из контекстного меню окна графиков командами **Добавить - Динамические звенья**. По этой команде добавляется график выбранного звена, как кривая разгона, реакция на единичное воздействие. Началу кривой разгона процесса соответствует левая граница времени тренда. Концу кривой разгона – правая граница. Коэффициент усиления звена равен по умолчанию единице. Постоянная времени звена задается как 1/10 от видимого интервала времени. В дальнейшем, границы и настройки звена можно откорректировать в режиме настроек. Определены следующие типы звеньев: *апериодическое, двойное апериодическое, пропорциональное, интегрирующее, дифференцирующее, колебательное, консервативное*. Дополнительные настройки звена:

- **Начало** - начало кривой разгона;
- **Конец** - конец кривой разгона;
- **Тип** - тип звена;
- **Параметры K, T.**

Линейный график состояний позволяет отображать условные сообщения в графическом виде. **График событий** позволяет отображать безусловные сообщения в графическом виде.

Форматеры позволяют задавать произвольное форматирование подписей осей. Форматтер можно задавать в виде строки. При этом первый символ строки идентифицирует форматтер,

а последующие - строку формата. В строке формата могут быть произвольные символы - если они не распознаны как управляющие, то они просто вставляются в выходную строку.

Форматер значений - первый символ ! идентифицирует *форматтер значений*, строка формата показывает кол-во нулей после запятой. Например: !F2 – отформатирует значения в виде 12.00, !F9 – отформатирует значения в виде 12.000000000.

Первый символ \$ идентифицирует *форматтер даты/времени*, строка формата является стандартной для форматирования времени: *y* - год, *M* - месяц, *d* - день, *H* – час в 24-формате, *h* – час в 12-формате, *m* - минуты, *s* - секунды, *f* - миллисекунды. Например: \$yyyy.MM.dd HH:mm:ss.fff отформатирует значения в виде 2008.08.12 23:33:06.444, \$dddd.MMMM.yy отформатирует значения в виде *понедельник 12.январь.08*, \$MM:ss отформатирует значения в виде 05:34.

Первый символ % идентифицирует *форматтер относительной шкалы* (автоматическое приведение к процентам). Строка формата может быть: %*min*#*max*, где *min* - значение, соответствующее 0%, а *max* - значение, соответствующее 100%. %*min*#*max*#*format*, то же самое, но *format* - опциональный параметр: формат относительного значения (количество нулей после запятой). Например: %0#100 – отформатирует шкалу, где 0% = 0, 100%=0, %-40#70 – отформатирует шкалу, где 0% = -40, 100%=70, %-50#90#4 – отформатирует шкалу, где 0% = 50, 100%=90, при этом значения будут выводиться с 4 знаками после запятой - 76.4563%.

Первый символ & идентифицирует *форматтер перечислений*, строка формата может быть следующей 0=*значение1*,1=*значение2*,3=*значение3*. При этом на оси значений будут подписи *значение1*, *значение2*, *значение3* вместо 1,2,3.

Вспомогательное окно графиков предназначено для анализа схожих ситуаций. Это окно доступно только в продвинутом уровне сложности. Для его отображения следует потянуть за ресайзер, расположенный между тулбаром и основным окном графиков. У вспомогательного окна графиков текущий интервал времени не связан с интервалом времени тренда, что дает возможность сравнивать текущую ситуацию с ситуацией в прошлом.

Таблицы

К таблицам относятся: таблица перьев (на панели перьев в легенде), таблица разметки (на панели разметки в легенде), таблица фильтров (на панели выбора сообщений в легенде), таблица сообщений (на панели сообщений), минилегенда. Все таблицы объединяет одинаковый пользовательский интерфейс и одинаковый способ настройки. Разница между таблицами и одноименными панелями следующая: панели кроме таблицы могут содержать дополнительные элементы, например, тулбар.

В таблицах реализована следующая общая функциональность:

- *Сортировка по одному или нескольким столбцам.* Сортировка может быть по возрастанию или убыванию. Переключение направление сортировки осуществляется кликом по заголовку столбца в следующем порядке: сортировка по убыванию / сортировка по возрастанию / отключение сортировки. Если используется множественная сортировка – то сначала сортировка осуществляется по первому столбцу, а если значения в этом столбце совпадают, то проверяются остальные столбцы в порядке возрастания индекса сортировки. Индекс сортировки отображается в заголовке столбца;
- *Переименование столбцов.* При наведении мыши на заголовок столбца появляется кнопка переименования столбца. При клике на эту кнопку открывается диалог задания имени столбца;
- *Изменение размера столбцов.* Изменение размера столбца осуществляется перемещением мыши при нажатой кнопке на ресайзере столбца;
- *Выбор столбцов.* Выбор видимых столбцов осуществляется из контекстного меню таблицы значений;
- *Перемещение столбцов.* Перемещение столбцов осуществляется перемещением мыши при нажатой кнопке на заголовке столбца.

Настройки таблиц:

- *Автовысота строк.* При включенной автовысоте – высота строки в таблице зависит от высоты содержимого. Например высота строки будет подгоняться под высоту шрифта или под многострочный текст. При выключенной автовысоте появляется настройка *Высота строк*, которая задает фиксированную высоту;

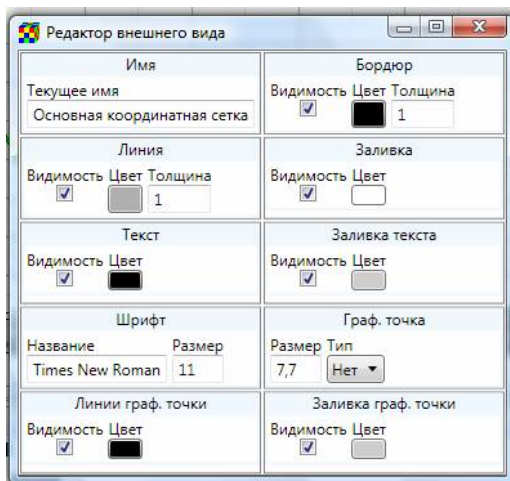
- *Контекстное меню.* Позволяет полностью отключить контекстное меню таблицы;
- *Множественная сортировка.* Позволяет включать/отключать сортировку по нескольким столбцам;
- *Поля записи.* Позволяет задавать поля каждой записи в таблице в пикселах. Поля задаются или одним значением или через запятую: левое, верхнее, правое, нижнее. *Поле* – означает пустое пространство;
- *Размер шрифта.* Позволяет задавать размер шрифта записей и заголовка таблицы;
- *Толщина границы записи.* Позволяет задавать толщину окантовки записи. Значение толщины задается аналогично полям записи;
- *Управление переименованием.* Позволяет отключать переименование столбцов;
- *Управление размером столбцов.* Позволяет отключать изменение размера столбцов;
- *Управление сортировкой.* Позволяет отключать изменение сортировки столбцов;
- *Управление столбцами.* Позволяет отключать любые действия со столбцами, в том числе вышеперечисленные.

Также управление столбцами можно отключить из права оператора для журнала *Изменение внешнего вида*.

Диалоговые окна

Диалог настройки внешнего вида разметки позволяет редактировать следующие характеристики:

- Пользовательское имя;
- Внешний вид бордюра линий (окантовки линий);
- Внешний вид линий;
- Внешний вид заливки пера;
- Внешний вид текста;
- Внешний вид фона текста;
- Шрифт текста;
- Тип графической точки;
- Внешний вид линий графической точки;
- Внешний вид заливки графической точки.

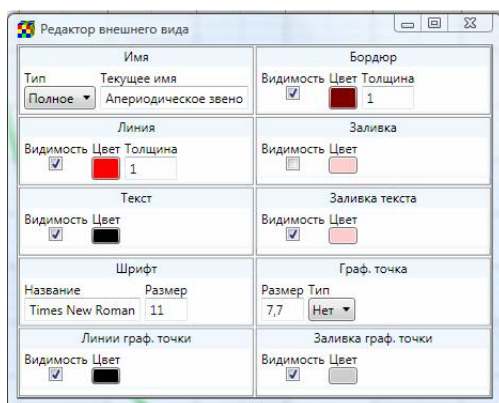


Возле некоторых характеристик имеются кнопки. Цвет кнопки соответствует цвету характеристики. При нажатии на кнопку вызывается диалог выбора цвета.

Диалог настройки внешнего вида пера позволяет редактировать следующие характеристики:

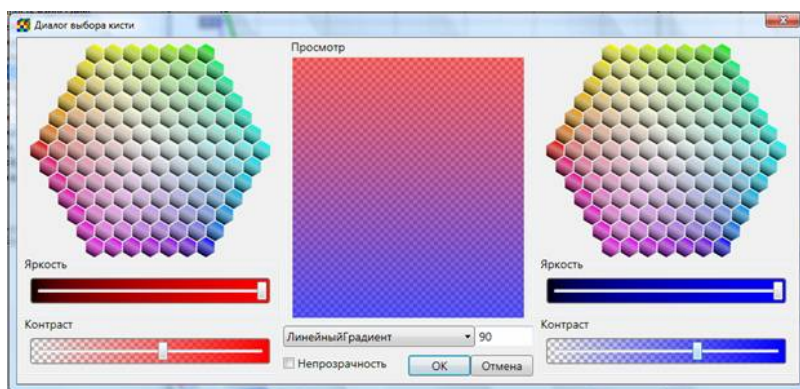
- Тип названия и пользовательское имя;
- Внешний вид бордюра линий (окантовки линий);
- Внешний вид линий;
- Внешний вид заливки пера;
- Внешний вид текста;
- Внешний вид фона текста;
- Шрифт текста;

- Тип графической точки;
- Внешний вид линий графической точки;
- Внешний вид заливки графической точки.



Возле некоторых характеристик имеются кнопки. Цвет кнопки соответствует цвету характеристики. При нажатии на кнопку вызывается диалог выбора цвета.

Диалог выбора цвета позволяет выбрать нужный цвет.

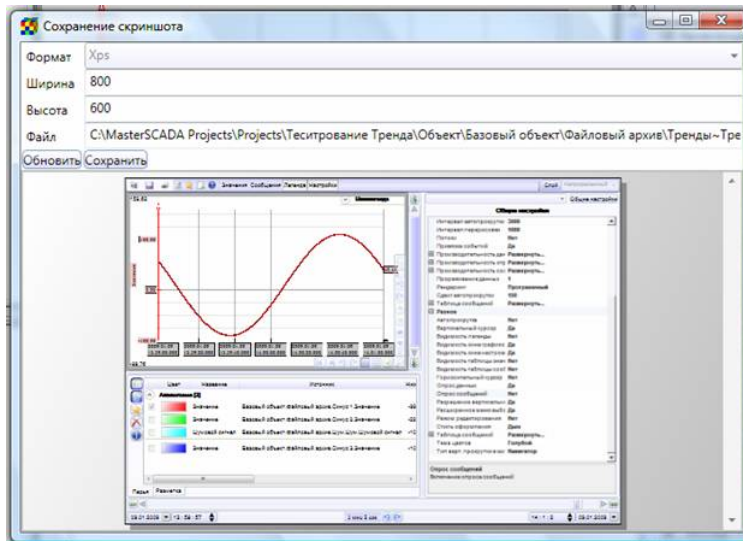


Для выбора цвета сначала следует выбрать тип кисти:

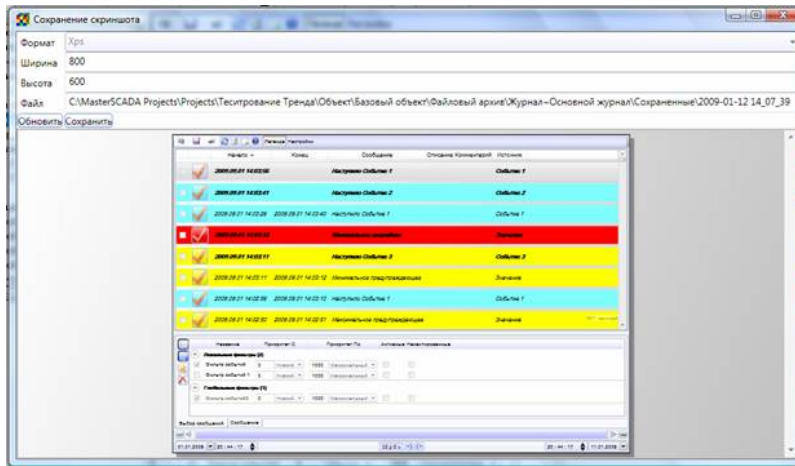
- Солидная (один цвет);
- Линейный градиент (смесь двух цветов вдоль линии);
- Радиальный градиент (смесь двух цветов по радиусу круга).

Если выбрана солидная кисть – то следует выбрать нужный оттенок цвета из гексагональной области, затем установить яркость цвета и контраст, а также поставить галочку – прозрачен цвет или нет. Если выбрана градиентная кисть (линейная или радиальная) – следует выбрать таким же способом второй цвет кисти. Также для линейного градиента следует задать угол его распространения.

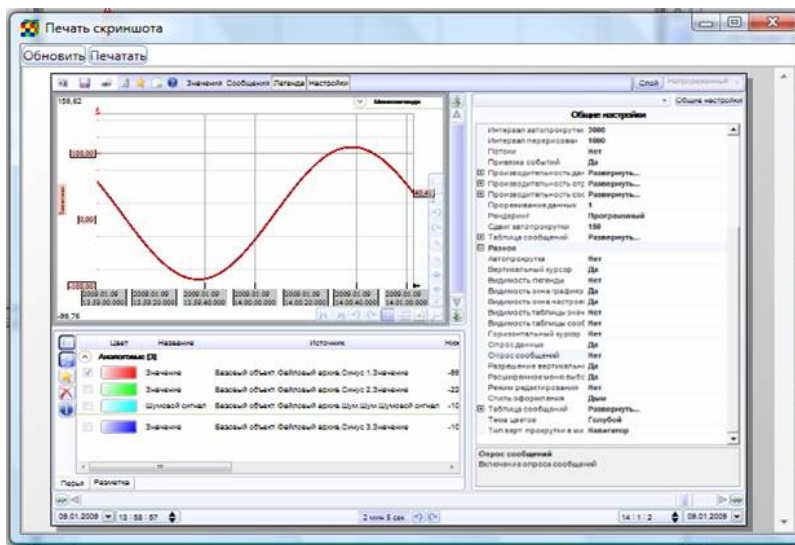
Диалог сохранения/печати снимка экрана предназначен для сохранения в файл или вывода на печать заданной области экрана. В режиме сохранения снимка в файл - диалог имеет вид как на картинке ниже.



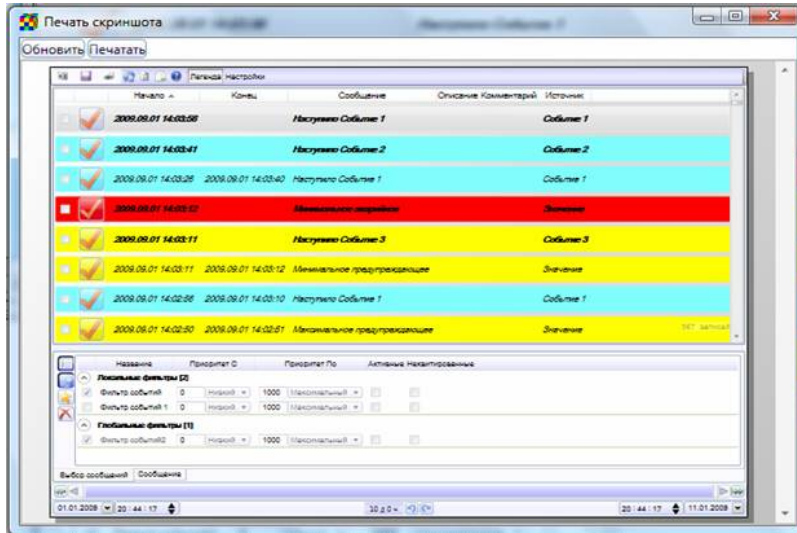
Или следующий вид.



Кнопка **Обновить** производит обновление зоны предварительного просмотра. Кнопка **Сохранить** сохраняет снимок в файл с заданными параметрами. В режиме вывода снимка на печать – диалог имеет следующий вид:



Или:



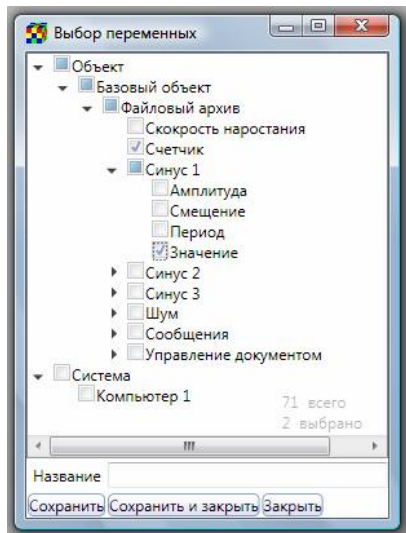
Кнопка *Обновить* производит обновление зоны предварительного просмотра. Кнопка *Печатать* открывает стандартный диалог печати.


В режиме вывода сообщений на печать – *Диалоговое окно печати сообщений* имеет следующий вид:

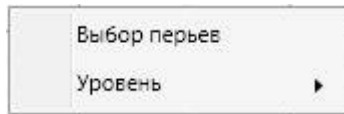


Кнопка *Печатать* открывает стандартный диалог печати.

Диалог выбора перьев предназначен для добавления или удаления перьев.



Данный диалог можно вызвать нажатием на кнопку  выбора перьев. Эта кнопка расположена в тулбаре тренда, а также в тулбаре панели перьев у легенды. Также этот диалог можно вызывать из контекстного меню окна графиков:



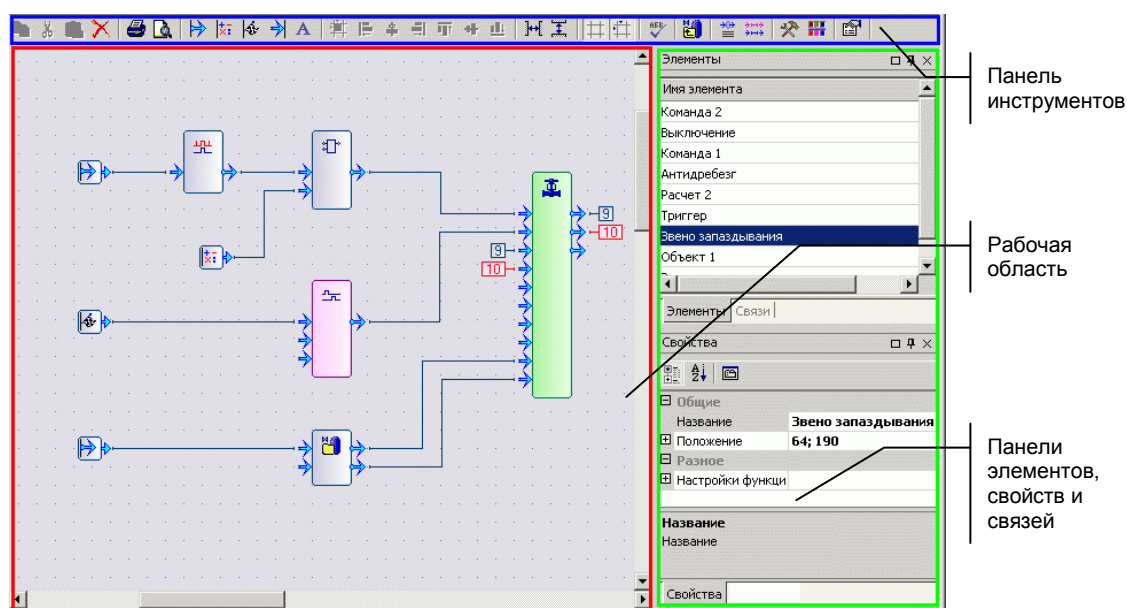
Содержимое диалогового окна представляет собой дерево объектов и дерево системы MasterSCADA, в котором помечены выбранные перья. Объекты, у которых выбраны все перья – помечены галочкой, а объекты, у которых дочерние перья выбраны частично – помечены другим символом или нарисованы серым цветом (в зависимости от темы Windows). Внизу диалогового окна находится фильтр названия, при вводе в него текста – деревья будут содержать только те элементы, которые имеют в составе названия строку фильтра.

Редактор схем функциональных блоков

Редактор предназначен для создания схем функциональных блоков. Схема функциональных блоков описывает порядок выполнения блоков и связи по передачи данных от выхода одного блока к входам других блоков.

Схема функциональных блоков представляет собой графическое отображение набора функциональных блоков, принадлежащих объекту. Каждому объекту может принадлежать не более одной схемы.

Пользовательский интерфейс




Рабочая область

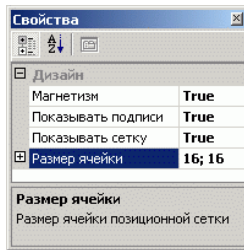
Рабочая область предназначена для отображения и редактирования схемы функциональных блоков. На схеме отображаются все переменные, функциональные блоки и объекты, входящие в состав текущего объекта.

Элементы схемы функциональных блоков:

- Функциональные блоки;
- Визуальные функциональные блоки;
- Объекты из дерева объектов Проекта, подчиненные объекту, которому принадлежит схема;
- Группы переменных;
- Переменные.


Свойства рабочей области

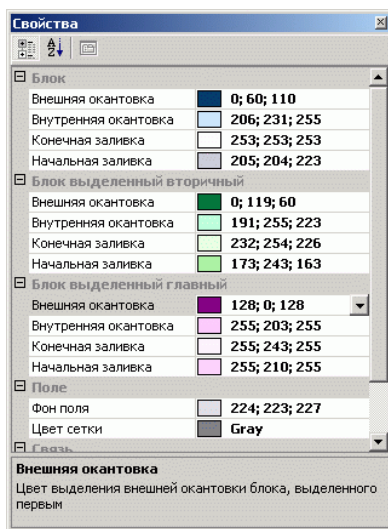
Свойства схемы отображаются на панели **Свойства**. Существует два режима настройки: настройка основных свойств и настройка цветов схемы. Для редактирования основных свойств, необходимо щелкнуть мышью по иконке  панели инструментов, либо, щелкнув мышью по пустому месту схемы, перейти на панель **Свойства**.





Свойства схемы:

- Магнетизм - выравнивает элементы по узлам сетки;
- Показывать подписи - показывает подписи имен элементов схемы, названия входов и выходов функциональных блоков;
- Показывать сетку - включает отображение сетки на схеме;
- Размер ячейки - устанавливает высоту и ширину ячейки сетки.

Для открытия панели **Свойства** в режиме настройки цветов схемы, необходимо щелкнуть мышью по иконке .



Отображение элементов на схеме

Наименование	Внешний вид	Описание
Объекты		Изображения дочерних объектов. Переменные Проекта, входящие в состав дочернего объекта, отображаются в виде входов и выходов объекта (Входы: <i>Значения</i> ; Выходы: <i>Команды</i> , <i>Расчеты</i> , <i>События</i>).
Функциональные блоки		В режиме отображения подписей функциональные блоки вместо изображения входов/выходов выводят их короткие имена.

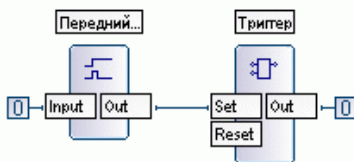
Наименование	Внешний вид	Описание
Группы переменных		
Команда		присутствует только в MSRT (исполнительная система MasterSCADA)
Расчёт		
Событие		присутствует только в MSRT (исполнительная система MasterSCADA)
Значение		присутствует только в MSRT (исполнительная система MasterSCADA)

Отображение связей

Существует три типа связи: связь с элементом дочернего объекта, внутренняя связь и внешняя.

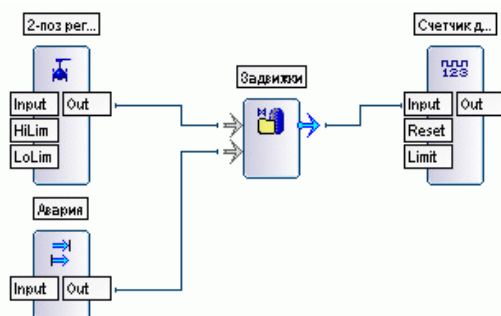
Внутренняя связь осуществляется между элементами, которые находятся внутри текущего объекта на первом уровне вложенности. На схеме отображаются одним из следующих способов:

- Линиями, если выход находится левее входа;
- Цифрами (маркерами), если выход находится правее входа.

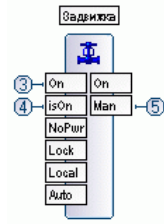


Связь с элементом дочернего объекта осуществляется между элементом текущего объекта и элементом дочернего объекта, и выглядит на схеме аналогично внутренней связи. При этом, дочерний объект отображается в виде блока, имеющего входы и выходы, которые представляют входы и выходы вложенных в него элементов. Входы/выходы дочернего объекта могут отображаться одним из двух способов:

- **Серым цветом** - если связь установлена со входом/выходом функционального блока, входящего в состав дочернего объекта (или любого другого объекта, принадлежащего дочернему);
- **Синим цветом** - если связь установлена с любой из переменных, принадлежащих непосредственно дочернему объекту.




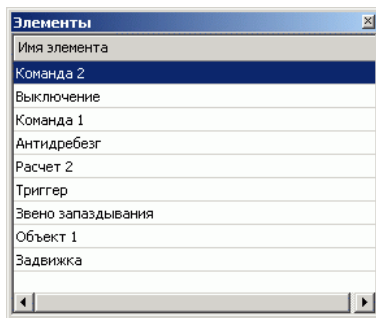
Внешняя связь осуществляется между элементом текущего объекта и элементом, не входящим ни в данный объект, ни в любой из дочерних. В этом случае связь отображается цифрой помещенной в круг.



Виды и функции панелей элементов, связей и свойств


Панель **Элементы** служит для отображения всех элементов схемы. Если щелкнуть мышью по имени элементу в панели, то он станет выделенным в рабочей области схемы. Активировать панель **Элементы** можно одним из следующих способов:

- Щелкнуть мышью по панели;
- Воспользоваться панелью инструментов, иконкой , или соответствующим пунктом меню.



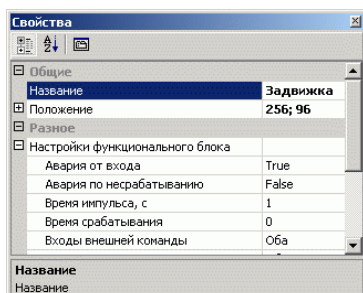
Панель **Связи** служит для отображения всех связей между элементами схемы. Если щелкнуть мышью по связи в панели, то она станет выделенной в рабочей области схемы.




Активировать панель **Связи** можно одним из следующих способов:

- Щелкнуть мышью по панели;
- Воспользоваться панелью инструментов, иконкой , или соответствующим пунктом меню.

Источник	Выход	Приемник	Вход
Команда 2	Команда 2	Антидребезг	Вход
Выключение	Выключение	Звено запазд	Вход
Команда 1	Команда 1	Объект 1	Значение 1
Антидребезг	Выход	Триггер	Включить
Расчет 2	Расчет 2	Триггер	Отключить
Триггер	Выход	Задвижка	Вход Откры
Звено запазд	Выход	Задвижка	Вход Закры
Объект 1	Команда 1	Задвижка	Местный ре
Объект 1	Расчет 1	Задвижка	Автомат
Задвижка	Выход Откры	Задвижка	Открыт

Панель **Свойства** служит для отображения и редактирования свойств выделенного в схеме элемента. Для открытия панели **Свойства**, необходимо выделить необходимый элемент и перейти на эту панель.




























Кнопка  выводит список свойств элемента по категориям, кнопка  выводит список свойств элемента, отсортированный по алфавиту. При нажатии кнопки  открывается страница свойств элемента. Кнопка становится доступной только в том случае, если выделенный элемент является функциональным блоком или переменной Проекта.

Свойства *Название* и *Положение* являются общими для всех элементов. Функциональные блоки имеют также категорию *Настройки* функционального блока, в которой можно произвести его настройку.

Панель инструментов

Кнопки, используемые для получения быстрого доступа к наиболее часто используемым функциям, описаны в списке ниже.

	Добавление объекта на схему
	Вставка функционального блока <i>Повторитель</i> (имеется только в MasterLogic-системе)
	Удаление выделенного элемента
	Добавление элемента <i>Команда</i>
	Добавление элемента <i>Расчёт</i>
	Добавление элемента <i>Событие</i>
	Добавление элемента <i>Значение</i>
	Вставка текста
	Выравнивание выделенного элемента по сетке
	Выравнивание группы выделенных элементов по левой границе
	Выравнивание группы элементов по общей вертикальной оси
	Выравнивание группы выделенных элементов по правой границе
	Выравнивание группы выделенных элементов по верхней границе
	Выравнивание группы элементов по общей горизонтальной оси
	Выравнивание группы выделенных элементов по нижней границе
	Распределение группы выделенных элементов по горизонтали. Выравнивается расстояние между вертикальными осями элементов
	Распределение группы выделенных элементов по вертикали. Выравнивается расстояние между горизонтальными осями элементов
	Отображение сетки на рабочей области
	Включение магнетизма
	Отображение подписей элементов и коротких имён входов и выходов функциональных блоков
	Открытие схемы родительского объекта
	Активизация панели <i>Элементы</i>
	Активизация панели <i>Связи</i>
	Активизация панели <i>Свойства</i> для настройки свойств мнемосхемы
	Активизация панели <i>Свойства</i> в режиме настройки цветов схемы

Разработка схемы функциональных блоков

Общие методы и способы разработки

При создании схемы функциональных блоков, на рабочую область автоматически добавляются все элементы, которые содержатся в дереве текущего объекта, в соответствии с порядком вычисления элемента. Добавлять новые элементы на схему можно одним из следующих способов:

- Вставить элемент из палитры. Выбор нужного элемента (например, функционального блока) в палитре элементов осуществляется нажатием левой клавиши мыши на элементе, для сброса - нажмите правую клавишу. Выбранный элемент перемещается мышью в нужное место в схеме и вставляется нажатием левой клавиши. Одновременно элемент добавится в дерево Объекта в соответствующее место, согласно положению на схеме;
- При помощи соответствующих иконок на панели инструментов;
- Вставить элемент в дерево Объекта. Элемент вставляется на схему в зависимости от его положения в дереве Объекта.


Существует два способа выделения элементов: множественное и единичное. Единичное выделение осуществляется одним из способов:

- Щелкнуть мышью по элементу в рабочей области;
- Выделить необходимый элемент на панели *Элементы*.

Множественное выделение осуществляется одним из способов:

- Щелкните левой клавишей мыши на требуемом элементе, удерживая клавишу *Ctrl*. Этот элемент изменит свой цвет (если уже были выбраны элементы, то этот элемент к ним добавится);
- Нажать левую клавишу мыши на пустом месте мнемосхемы, затем не отпуская ее тянуть рамку, пока она не охватит все нужные для выделения элементы. Отпустить левую клавишу мыши.

При *множественном выделении* элемент, который был выделен первым становится главным среди выделенных, его координаты берутся за основу при выполнении операций выравнивания. При щелчке левой кнопки мыши в рабочей области схемы, с нажатой клавишей *Alt*, выводится список всех элементов находящихся под курсором мыши.

- ❖ **Важно!** Удаление элементов из схемы приведет к их удалению из дерева Объектов. Для удаления элемента, необходимо выделить его на схеме, затем нажать на иконку  панели инструментов, либо нажать клавишу Delete.

Перемещение элементов на схеме приводит к их перемещению в дереве Объектов и, следовательно, к изменению порядка вычисления элементов. Для перемещения элементов необходимо выделить их на рабочей области, и удерживая левую кнопку мыши перетащить в нужное положение. Также можно изменить положение элемента на панели *Свойства*, в категории *Положения*.


Связь устанавливается между выходом одного элемента и входом другого элемента. Вход может иметь только одну связь с каким-либо выходом. Выход может иметь связь со сколь угодно большим числом *Входов*.

Связь может быть установлена следующими способами:

- В дереве *Проекта*. Для установления связи необходимо перетащить левой кнопкой мыши одну переменную на другую. Этим способом можно установить связь между элементами схем разных объектов;
- В *рабочей области* схемы. Для этого нужный выход перетащите на необходимый вход;
- Перетаскивание *Вход* или *Выход* деревьев проекта на вход или выход элемента на схеме. Этим способом также можно установить связь между элементами схем разных объектов. В текущей версии редактора не разрешается создавать связи путем перетаскивания входа/выхода из схемы в деревья проекта.

В программе допускается устанавливать связь между переменными разных типов. При этом если в результате этой связи происходит потеря точности, выдается соответствующее пре-

дупреждение. Каждый тип данных имеет стрелки разных цветов. По такому же принципу устроено и отображение связей разных типов. Если установлена связь между переменными разных типов, то связь отображается пунктиром, причем первая часть пунктира приобретает цвет выхода, а последняя часть - цвет входа. Все имеющиеся связи отображаются на панели *Связи*.

Свойства связей отражаются на панели *Свойства*. Для удаления связи необходимо выделить ее в рабочей области, либо на панели *Связи*, затем щелкнуть по иконке  на панели инструментов.

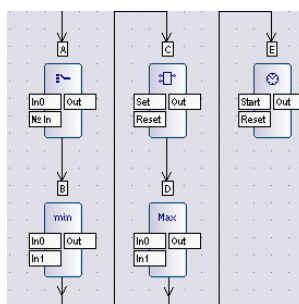
Работа в режиме исполнения

Элементы на схеме предоставленной ниже располагаются соответственно порядку вычисления циклов исполнительской системы. Порядок вычисления сверху вниз слева направо, то есть элемент, у которого значение координаты X меньше предыдущих, вычисляется раньше. В случае если элементы имеют одинаковые координаты по X, то первым вычисляется тот, у которого меньше значение координаты Y. Стрелками показан порядок вычисления элементов.

На схеме в *режиме Исполнения* можно редактировать значения переменных. Для этого, после щелчка правой кнопкой мыши на значении, в появившемся окне *Ввод значения*, введите новое значение переменной. Вид этого окна зависит от типа переменной. Например, если переменная имеет перечислимый тип, то значение выбирается из выпадающего списка и т.п.

Вид дерева Объектов

Вид схемы



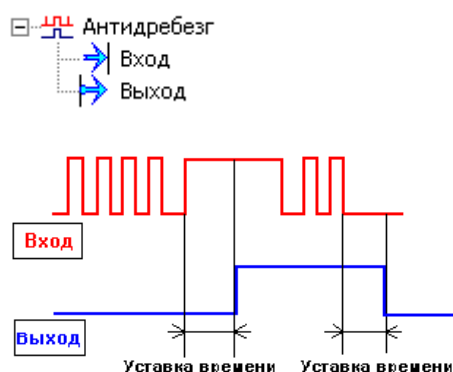
Палитра функциональных блоков

Обработка сигналов

Антидребезг

Сигнал со *Входа* проходит на *Выход*, если после последнего изменения данных на *Входе* прошло не меньше заданной уставки времени. Уставка времени задается на странице *Свойства* функционального блока.

Вид функционального блока в дереве объекта:

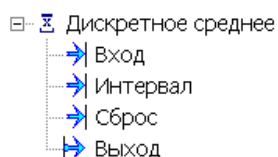


Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Дискретное среднее

В течение текущего *Интервала времени* на *Выход* подается среднее значение из пришедших на *Вход* за предыдущий *Интервал*. (Среднее значение рассчитывается как отношение интеграла за период к этому периоду). Если произошел переход Сброс из "Ложь" в "Истина", то расчет начинается заново. Изменение Интервала равносильно переходу Сброс из "Ложь" в "Истина"

Вид функционального блока в дереве объекта:



Дисциплина опроса должна быть периодической. Хорошая точность достигается, когда *Период опроса* по крайней мере в 20 раз меньше *Интервала*.

Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

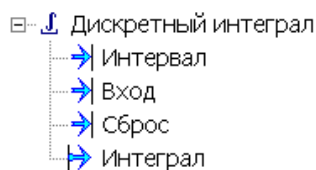
Поддерживаемый тип опроса – периодический. Восстановление при рестарте - не требуется.

Дискретный интеграл

Интеграл рассчитывается по формуле прямоугольников. В течение текущего *Интервала времени* на *Выход* подается интеграл значений, пришедших на *Вход* за предыдущий *Интервал*.

вал. Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*.

Вид функционального блока в дереве объекта:



Дисциплина опроса должна быть периодической. Хорошая точность достигается, когда *Период опроса* по крайней мере в 20 раз меньше *Интервала*.

Входы и выходы:

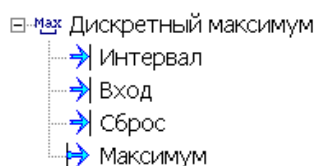
	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

Поддерживаемый тип опроса – периодический. Восстановление при рестарте - не требуется.

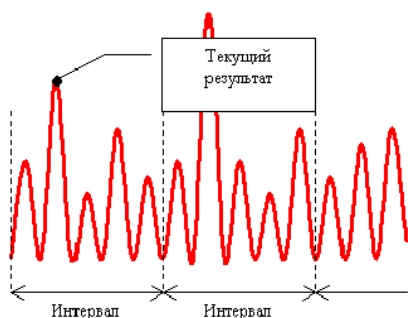
Дискретный максимум

В течение текущего *Интервала времени* на *Выход* подается максимальное из значений, которые приходили на *Вход* в течение предыдущего *Интервала времени*. Переход *Сброс* из *Ложь* в *Истина* дает команду начать расчет заново.

Вид функционального блока в дереве объекта:



Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*.



Входы и выходы:

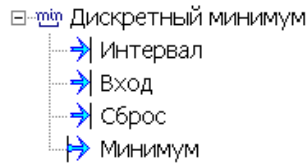
	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Сброс	Логический
Выходы	Максимум	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

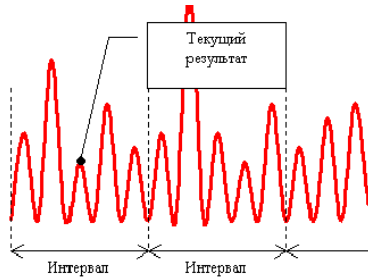
Дискретный минимум

В течение текущего *Интервала времени* на *Выход* подается минимальное из значений, которые приходили на *Вход* в течение предыдущего *Интервала времени*. Переход *Сброс* из *Ложь* в *Истина* дает команду начать расчет заново.

Вид функционального блока в дереве объекта:



Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*.



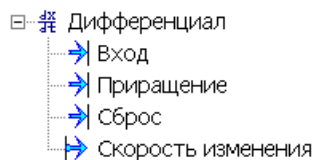
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Сброс	Логический
Выходы	Минимум	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Дифференциал

Рассчитывает производную dx/dt входного сигнала по времени. Вид функционального блока в дереве объекта:



Промежуток времени dt задается на входе *Приращение*. dx - изменение параметра *Вход* за этот промежуток. Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается сначала. Изменение *Приращения* равносильно переходу *Сброс* из *Ложь* в *Истина*.

Входы и выходы:

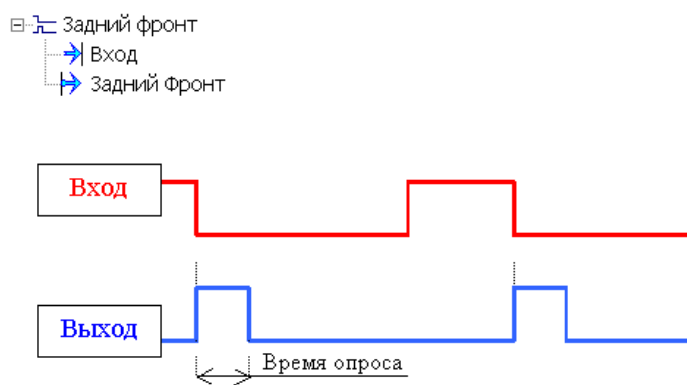
	Обозначение	Тип
Входы	Вход	Вещественный
	Приращение	Вещественный
	Сброс	Логический
Выходы	Скорость изменения	Вещественный

Поддерживаемый тип опроса – периодический. Восстановление при рестарте - не требуется.

Задний фронт

При переходе значения *Входа* из *Истина* в *Ложь* на выход подается *Истина*. При следующем опросе блока значение выхода переходит в *Ложь* и ожидается новый переход входа из *Истина* в *Ложь*.

Вид функционального блока в дереве объекта:



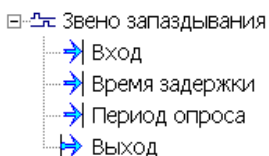
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Логический
	Задний фронт	Логический
Выходы		

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Звено запаздывания

Запоминает значение *Входа* и передает его на *Выход* позже текущего времени на *Время задержки*. Вид функционального блока в дереве объекта:



Реально запоминаются значения через каждые *Период опроса секунд*. Среди них отыскивается то, у которого астрономическое время, когда его запомнили меньше текущего времени на *Время задержки*. Так как функция запоминает много значений, необходимо соблюдать условие: $(\text{Время задержки} / \text{Период опроса}) < 15000$.

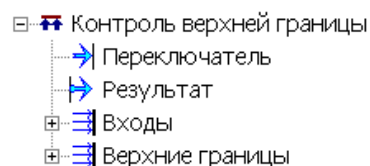
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Время задержки	Вещественный
	Период опроса	Вещественный
	Выход	Вещественный
Выходы		

Поддерживаемый тип опроса – периодический. Восстановление при рестарте - не требуется.

Контроль верхней границы

Следит за значениями нескольких параметров, не больше ли они заданных чисел. Вид функционального блока в дереве объекта:



Число контролируемых величин (тип *Целое*) может принимать значение от 1 до 128. Если на вход *Переключатель контроля по И / по ИЛИ* подано значение *Истина*, на выходе Q будет *Истина* только если значения всех контролируемых величин станут выше своих границ. Если на вход *Переключатель контроля по И / по ИЛИ* подано значение *Ложь*, на выходе Q

будет *Истина* если значение одной из всех контролируемых величин станет выше своей границы.

Входы и выходы:

	Обозначение	Тип
Входы	Переключатель	Логический
	Значение 1	Вещественный
	Верхняя граница 1	Вещественный

	Значение N	Вещественный
Выходы	Верхняя граница N	Вещественный
	Результат	Логический

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Контроль диапазона

Следит за значениями нескольких параметров, на предмет соответствия их заданному интервалу. *Число контролируемых величин* (тип *Целое*) может принимать значение от 1 до 128. Если на вход *Переключатель* контроля по *И/ИЛИ* подано значение "*Истина*", на выходе будет "*Истина*" только если значения всех контролируемых величин выйдут за пределы установленного интервала. Если на вход *Переключатель* контроля по *И/ИЛИ* подано значение "*Ложь*", на выходе будет "*Истина*" если значение одной из всех контролируемых величин выйдут за пределы установленного интервала.

Входы и выходы:

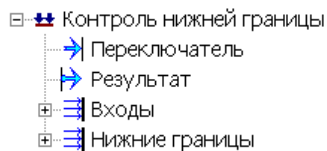
	Обозначение	Тип
Входы	Переключатель	Логический
	Вход 1	Вещественный
	Верхняя граница 1	Вещественный
	Нижняя граница 1	Вещественный

	Вход N	Вещественный
	Верхняя граница N	Вещественный
Выходы	Нижняя граница N	Вещественный
	Результат	Логический

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Контроль нижней границы

Следит за значениями нескольких параметров, не меньше ли они заданного числа. Вид функционального блока в дереве объекта:



Число контролируемых величин (тип *Целое*) может принимать значение от 1 до 128. Если на вход *Переключатель* контроля по *И / по ИЛИ* подано значение *Истина*, на выходе Q будет *Истина* только если значения всех контролируемых величин станут ниже своих границ. Если на вход *Переключатель* контроля по *И / по ИЛИ* подано значение *Ложь*, на выходе Q будет *Истина* если значение одной из всех контролируемых величин станет ниже своей границы.

Входы и выходы:

	Обозначение	Тип
Входы	Переключатель	Логический
	Значение 1	Вещественный
	Нижняя граница 1	Вещественный

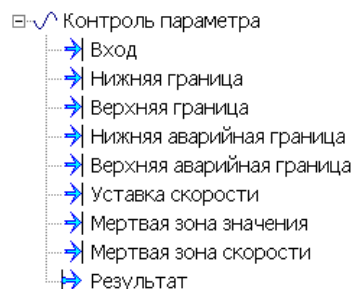
	Значение N	Вещественный

	Обозначение	Тип
Выходы	Нижняя граница N Результат	Вещественный Логический

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Контроль параметра

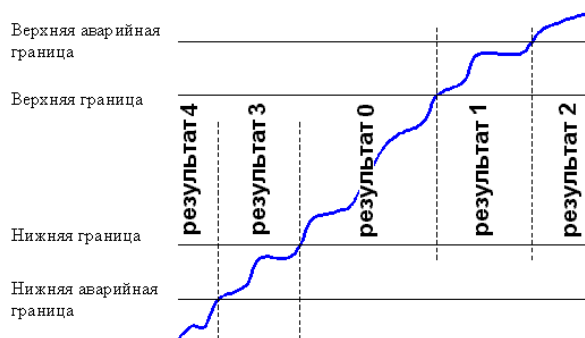
Следит за значением параметра. Вид функционального блока в дереве объекта:



Сначала производится контроль скорости изменения параметра. Скорость вычисляется как отношение изменения параметра к промежутку времени, за который произошло это изменение. Если значение скорости отличается от *Уставка Скорости* более, чем на значение *Зона нечувствительности* скорости, то есть $|Уставка\ скорости - Скорость| > Зона\ нечувствительности\ скорости$ то **Результат = 5**.

Если же скорость параметра попадает в нужный интервал, то производится контроль самого параметра:

- Если значение параметра *Вход* меньше *Верхней границы* и больше *Нижней*, то **Результат = 0**;
- Если значение параметра *Вход* меньше *Верхней аварийной границы*, но больше *Верхней*, то **Результат = 1**;
- Если значение параметра *Вход* больше *Верхней аварийной границы*, **Результат = 2**;
- Если значение параметра *Вход* меньше *Нижней границы*, но больше *Нижней аварийной*, то **Результат = 3**;
- Если значение параметра *Вход* меньше *Нижней аварийной границы*, **Результат = 4**.

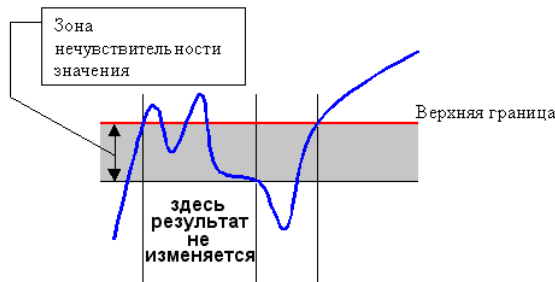


Дополнительные возможности: если параметр изменяется незначительно, результат остается прежним при выполнении следующих условий:

- Если значение параметра возрастает и переходит *Нижнюю аварийную границу* не более чем на величину *Зона нечувствительности значения*, результат остается равным 4;
- Если значение параметра возрастает и переходит *Нижнюю промежуточную границу* не более чем на величину *Зона нечувствительности значения*, результат остается равным 3;
- Если значение параметра убывает и переходит *Верхнюю границу* не более чем на величину *Зона нечувствительности значения*, результат остается равным 2;

- Если значение параметра убывает и переходит *Верхнюю промежуточную границу* не более чем на величину *Зона нечувствительности значения*, результат остается равным 1.

Это сделано во избежание *Дребезга* сигнала вблизи границ.



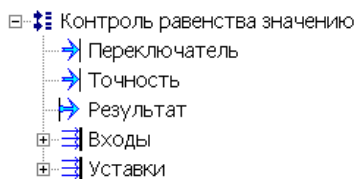
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Нижняя граница	Вещественный
	Верхняя граница	Вещественный
	Нижняя аварийная граница	Вещественный
	Верхняя аварийная граница	Вещественный
	Уставка скорости	Вещественный
	Зона нечувствительности значения	Вещественный
	Зона нечувствительности скорости	Вещественный
Выходы	Результат	Целый

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Контроль равенства значению

Следит за значениями нескольких параметров, на предмет равенства их заданному числу. *Контроль* срабатывает, если параметр отличается от заданного значения *Установка* более чем на *Точность*. Вид функционального блока в дереве объекта:



Число контролируемых величин (тип *Целое*) может принимать значение от 1 до 128. Если на вход *Переключатель контроля по И / по ИЛИ* подано значение *Истина*, на выходе будет *Истина* только если значения всех контролируемых величин будут отличаться от установленного. Если на вход *Переключатель контроля по И / по ИЛИ* подано значение *Ложь*, на выходе будет *Истина* если значение одной из всех контролируемых величин будет отличаться от установленного.

Входы и выходы:

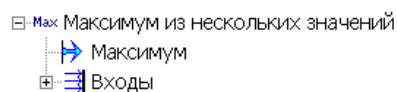
	ОБОЗНАЧЕНИЕ	ТИП
ВХОДЫ	Переключатель	Логический
	Значение 1	Вещественный
	Уставка 1	Вещественный

	Значение N	Вещественный
	Уставка N	Вещественный
	Точность	Вещественный
ВЫХОДЫ	Результат	Логический

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Максимум из нескольких значений

Выбирает максимальное значение из подключенных входов и подает на выход. Вид функционального блока в дереве объекта:



Настройки:

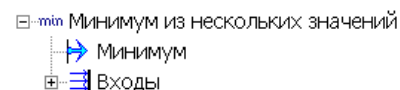
Свойство	Тип свойства	Значение
Число входов	Целое	От 2 до 255
Входы и выходы:		
	Обозначение	Тип
Входы	Вход 1	Вещественный

	Вход N	Вещественный
Выходы	Максимум	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Минимум из нескольких значений

Выбирает минимальное значение из подключенных входов и подает на выход. Вид функционального блока в дереве объекта:



Настройки:

Свойство	Тип свойства	Значение
Число входов	Целое	От 2 до 255
Входы и выходы:		
	Обозначение	Тип
Входы	Вход 1	Вещественный

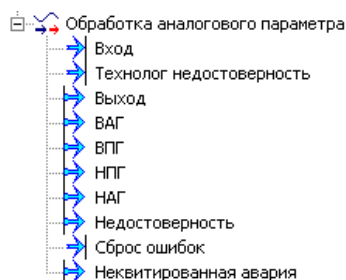
	Вход N	Вещественный
Выходы	Минимум	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

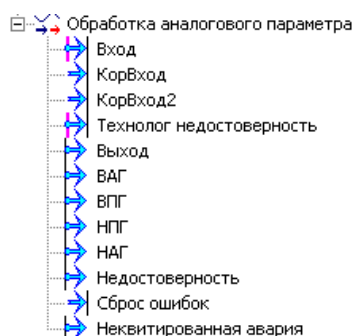
Обработка аналогового параметра

Функциональный блок служит для обработки в контроллере аналогового параметра, который поступает с нижнего уровня, с модулей ввода-вывода, и формирования внешнего вида отображения аналогового параметра на мнемосхеме. Значение, поступившее на *Вход* подвергается специальной обработке (фильтрация, корректировка, контроль достоверности и т.п) и затем выдается на выход. Дополнительные входы и выходы выводят информацию о нарушении аналоговым параметром аварийных и предупредительных границ, позволяют устанавливать программную недостоверность и т.п.

Вид ФБ без корректировки указан на рисунке ниже.



С коррективкой:



Поддерживаемый тип опроса – любой. Восстановление при рестарте - поддерживается.

Входы и Выходы ФБ:

Название	Тип	Назначение
Вход	Вещественный	Устанавливается связь с необходимым выходом модуля ввода-вывода, подключенного к контроллеру.
КорВход	Вещественный	Для расхода - задается значение давления, которое будет использоваться для расчета плотности (плотность участвует в расчете реального расхода воды или пара). Формула, используемая для расчета, приведена в справке. Для уровня - задается значение коэффициента k, участвующее в формуле расчета уровня.
КорВход2	Вещественный	Для расхода - задается значение температуры, которое будет использоваться для расчета плотности, (плотность участвует в расчете реального расхода воды или пара). Формула, используемая для расчета, приведена в справке. Для уровня - задается значение коэффициента b, участвующее в формуле расчета уровня.
Технолог достоверность	Логический	На вход подается программно рассчитанное логическое значение, определяющее технологическую достоверность.
Сброс ошибок	Логический	Позволяет выполнять программное квитирование аварийной или предаварийной ситуации.
Выход	Вещественный	Значение передается со <i>Входа</i> после специальной обработки. На странице свойств <i>Общие</i> настраивается диапазон изменения переменных.
ВАГ	Логический	Выход равен <i>Истина</i> , если параметр стал равен значению, которое больше верхней аварийной границы. Выход возвращается в <i>Ложь</i> , если параметр стал меньше верхней аварийной границы на величину, равную настройке <i>Зона нечувствительности</i> .
ВПГ	Логический	<i>Выход</i> равен <i>Истина</i> , если параметр стал равен значению, которое больше верхней предупредительной границы. <i>Выход</i> возвращается в <i>Ложь</i> , если параметр стал меньше верхней предупредительной границы на величину, равную настройке <i>Зона нечувствительности</i> .
НПГ	Логический	Выход равен <i>Истина</i> , если параметр стал равен значению, которое меньше нижней предупредительной границы. Выход возвращается в <i>Ложь</i> , если параметр стал больше нижней предупредительной границы на величину, равную настройке <i>Зона нечувствительности</i> .
НАГ	Логический	Выход равен <i>Истина</i> , если параметр стал равен значению, которое меньше нижней аварийной границы. <i>Выход</i> возвращается в <i>Ложь</i> , если параметр стал больше нижней аварийной границы на величину, равную настройке <i>Зона нечувствительности</i> .

Название	Тип	Назначение
Недостоверность	Логический	Выход равен <i>Истина</i> , если контролируемый параметр становится недостоверным в результате определения технологической либо аппаратной недостоверности.
Неквитированная авария	Логический	Выход равен <i>Истина</i> , если произошла какая-либо аварийная или предаварийная ситуация (сработала одна из аварийных, или предупредительных границ, наступила технологическая или аппаратная недостоверность и т.п.). При возвращении контролируемого параметра в норму, выход продолжает находиться в состоянии <i>Истина</i> , до тех пор, пока оператор не сбросит (квитирует) аварию с мнемосхемы или пока не произойдет программный сброс.

Значение параметра отображается на мнемосхеме в окне параметра цифрами черного цвета. Размерность параметра определяется назначенной шкалой для *Выхода* и располагается справа от значения. В нормальном состоянии параметра окно параметра индицируется голубым цветом (цвет может быть изменен в редакторе мнемосхем). Событие выхода значения параметра за предупредительную границу (верхнюю или нижнюю) индицируется желтым мигающим цветом окна параметра на фоне серого (цвета могут быть изменены в редакторе мнемосхем). Событие выхода значения параметра за аварийную границу (верхнюю или нижнюю) индицируется красным мигающим цветом на фоне серого (цвета могут быть изменены в редакторе мнемосхем). После съема мигания (квитирования) окно параметра начинает светиться ровным цветом, желтым или красным. Вход параметра в нормальные пределы сопровождается изменением цвета окна на голубой. Событие выхода параметра за какую-либо границу и возврат в нормальные пределы индицируется миганием голубого поля значения параметра.

Отображение	Поведение	Событие	Примечания
	Нормальное состояние аналогового параметра.	Мигание цвета поля только, если было нарушение какой-либо из границ и возврат в нормальное состояние без квитирования.	
	Нарушена предупредительная граница верхняя или нижняя.	Мигание цвета поля.	На соответствующем выходе (НПГ или ВПГ) будет единица.
	Нарушена аварийная граница верхняя или нижняя.	Мигание цвета поля.	На соответствующем выходе (ВАГ или НАГ) будет единица.
	Значение параметра недостоверно, отображено последнее достоверное значение, если значение неизвестно, то поле остается пустым.	Мигание цвета рамки.	Контроллер сформировал плохой признак качества переменной. Аппаратная недостоверность.
	Значение параметра недостоверно, отображено последнее достоверное значение, если значение неизвестно, то поле остается пустым.	Мигание цвета рамки.	Присутствует <i>Истина</i> на входе <i>Технологическая недостоверность</i> либо параметр признан недостоверным по скорости.
	Значение параметра может быть недостоверно. Параметр рассчитывается с использованием констант, а не реальных параметров.	Мигание цвета рамки.	Имеется внутренний признак <i>Низкая точность</i> .

Примечание: мигание цвета поля прекращается после того, как авария была квитирована. Поле вывода значения аналогового параметра является динамическим элементом. При нажатии на левую клавишу мыши открывается окно для получения информации (окно значения параметра). Элементы окна описаны в таблице ниже.

Поле	Содержание
Заголовок окна	Код KKS параметра или при его отсутствии - положение параметра в дереве объектов.
Наименование	Наименование параметра указанное в свойствах элемента в редакторе

Поле	Содержание
Значение	мнемосхем. В левом поле отображается текущее значение переменной, в правом - диапазон изменения.
ВАГ ВПГ НАГ НПГ	В соответствующих полях показывается текущее значение аварийных и предупредительных границ.
Диагностика Аппаратная недостоверность	При нажатии на данную кнопку раскрывается правая часть окна.
Технологическая недостоверность Недостоверность по скорости изменения Низкая точность Сброс ошибок	При возникновении соответствующей ситуации, ее индикатор становится красным. При нажатии на данную кнопку происходит квитирование аварийных и предаварийных ситуаций. Элемент на мнемосхеме перестает мигать. При нажатии на данную кнопку, выход ФБ <i>Неквитированная авария</i> становится равен <i>Ложь</i> .

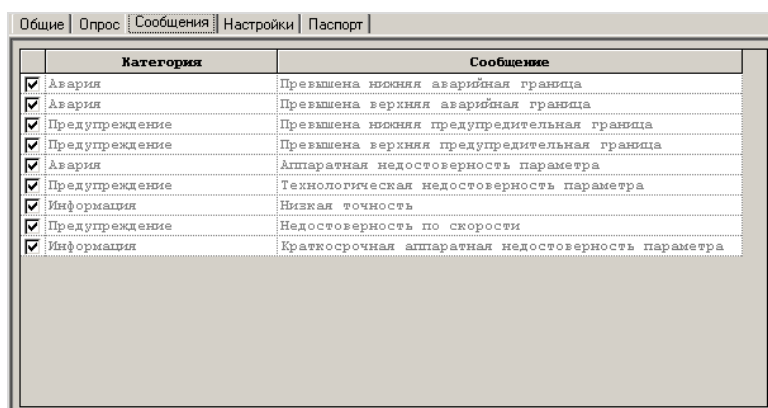
Страница свойств *Настройки* ФБ *Обработка аналогового параметра*.

Общие Опрос Сообщения Настройки Паспорт	
Контроль аппаратной достоверности	
Контрольное время аппаратной недостоверности	0
Контроль границ	
Верхняя аварийная граница	100
Верхняя предупредительная граница	95
Зона нечувствительности	1
Контролировать верхнюю аварийную границу	Истина
Контролировать верхнюю предупредительную границу	Истина
Контролировать нижнюю аварийную границу	Истина
Контролировать нижнюю предупредительную границу	Истина
Нижняя аварийная граница	0
Нижняя предупредительная граница	5
Контроль скорости	
Контролировать скорость изменения параметра	Истина
Приращение	1
Уставка скорости изменения	10
Контроль технологической достоверности	
Контролировать технологическую достоверность параметра	Истина
Корректировка параметра	
Значение К0 в коррекции расхода	1
Значение корректирующего параметра 1 по умолчанию	1
Значение корректирующего параметра 2 по умолчанию	1
Тип корректировки параметра	Расход воды
Разное	
Сокращенное наименование	
Фильтрация	
Использовать фильтр	Истина
Постоянная времени фильтра, с	1
Контроль аппаратной достоверности	
<input type="button" value="Применить"/> <input type="button" value="Отменить"/> <input type="button" value="Справка"/>	

Название	Назначение
Контрольное время аппаратной недостоверности	Сообщение об аппаратной недостоверности будет выдаваться после истечения времени заданного в поле. Значение в секундах.
Верхняя аварийная граница	Задается значение верхней аварийной границы.
Верхняя предупредительная граница	Задается значение верхней предупредительной границы.
Зона нечувствительности	Если значение параметра вернется из-за граничной уставки меньше чем на величину зоны нечувствительности, то признак пресечения границы не сбросится (Удалено - значение параметра вернется из-за граничной уставки меньше чем на величину зоны нечувствительности, то признак пресечения границы не сбросится).
Контролировать верхнюю аварийную границу	Определяется, нужно ли контролировать верхнюю аварийную границу.
Контролировать верхнюю предупредительную границу	Определяется, нужно ли контролировать верхнюю предупредительную границу.
Контролировать нижнюю аварийную границу	Определяется, нужно ли контролировать нижнюю аварийную границу.
Контролировать нижнюю предупредительную границу	Определяется, нужно ли контролировать нижнюю предупредительную границу.
Нижняя аварийная граница	Задается значение нижней аварийной границы.
Нижняя предупредительная граница	Задается значение нижней предупредительной границы.
Контролировать скорость изменения параметра	Определяется, нужно ли контролировать скорость изменения параметра. Если переменная изменилась в секунду на величину больше уставки, то формируется авария.
Приращение	Задается временной интервал для вычисления скорости измене-

Название	Назначение
Уставка скорости изменения	ния параметра. Чем выше значение этой величины тем меньше скачет значение скорости при малых колебаниях параметра. Задается величина уставки для формирования аварии по скорости изменения. Если переменная изменилась на данную величину за интервал указанный в поле приращение, то формируется авария.
Контролировать технологическую недостоверность параметра	Определяется необходимость ввести дополнительный вход <i>Технолог недостоверность</i> в функциональный блок. Возможность контролировать технологическую недостоверность параметра, вычисленную программно.
Тип корректировки параметра	Корректировки поддаются расходы воды и пара, а так же уровень. В данном поле выбирается, по какому алгоритму выполнять корректировку.
Значение k0 в коррекции расхода Значение корректирующего параметра 1 по умолчанию	Коэффициент сужающего устройства расходомера. Для расхода - задается значение давления P, которое будет использоваться для расчета плотности (плотность участвует в расчете реального расхода). Для уровня - задается значение коэффициента k, участвующее в формуле расчета уровня. Данные значение используется в случае, если на вход ФБ <i>КорВход</i> придет сигнал с плохим признаком качества. В этом случае срабатывает флаг <i>Низкая точность</i> .
Значение корректирующего параметра 2 по умолчанию	Для расхода - задается значение температуры, которое будет использоваться для расчета плотности. Для уровня - задается значение коэффициента b, участвующее в формуле расчета уровня. Данное значение используется в случае, если на вход ФБ <i>КорВход1</i> придет сигнал с плохим признаком качества. В этом случае срабатывает флаг <i>Низкая точность</i> .
Сокращенное наименование Использовать фильтр	Используется при составлении отчетов РОП. Требуется ли фильтровать полученное значение на входе функционального блока. В качестве фильтра используется апериодическое звено первого порядка.
Постоянная времени фильтра, с	Задается постоянная времени фильтра.

Страница свойств *Сообщения* ФБ *Обработка аналогового параметра*.

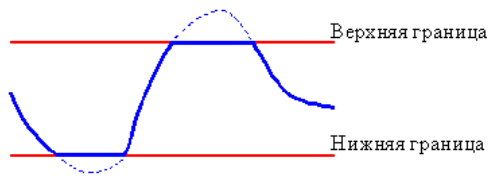
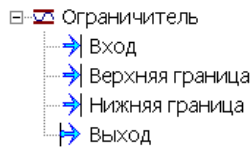


Текст	Событие
Превышена нижняя аварийная граница	Контролируемый параметр вышел за соответствующую границу
Превышена верхняя аварийная граница	
Превышена нижняя предупредительная граница	
Превышена верхняя предупредительная граница	
Аппаратная недостоверность параметра	Параметр, поступивший на вход ФБ, имеет недостоверный признак качества
Технологическая недостоверность параметра	Вход функционального блока <i>Технолог недостоверность</i> равен <i>Истина</i> .
Низкая точность	На входы функционального блока <i>КорВход</i> и <i>КорВход1</i> поступил сигнал недостоверным признаком качества.
Недостоверность по скорости	Превышена скорость изменения параметра
Краткосрочная аппаратная недостоверность параметра	Контроллер выдал аппаратную недостоверность параметра на время меньше, чем указано в настройках в поле <i>Контрольное время аппаратной недостоверности</i> .

Все сообщения выдаются по переднему фронту наступления события, канал вывода сообщения зависит от его категории. Категорию сообщения можно изменять на данной странице свойств.

Ограничитель

Обрезает амплитуду сигнала на входе *Верхней* и *Нижней* границами и подает результирующий сигнал на *Выход*. Вид функционального блока в дереве объекта:



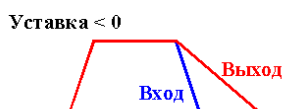
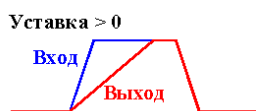
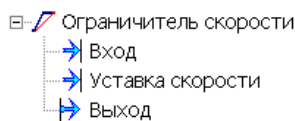
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Верхняя граница	Вещественный
	Нижняя граница	Вещественный
Выход	Выход	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Ограничитель скорости

Ограничивает скорость изменения сигнала. Если входной сигнал изменяется быстрее *Уставки скорости*, то на *Выход* подается сигнал, изменяющийся со скоростью уставки. Если значение *Уставки скорости* положительное, ограничивается скорость возрастания сигнала. Если отрицательное - ограничивается скорость убывания. Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется. Вид функционального блока в дереве объекта:

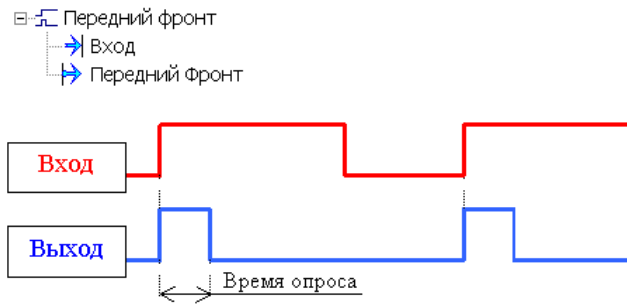


Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный двойной точности
	Уставка скорости	Вещественный двойной точности
Выход	Выход	Вещественный двойной точности

Передний фронт

При переходе значения Входа из Ложь в Истина на выход подается Истина. При следующем опросе блока значение выхода переходит в Ложь и ожидается новый переход входа из Ложь в Истина. Вид функционального блока в дереве объекта:



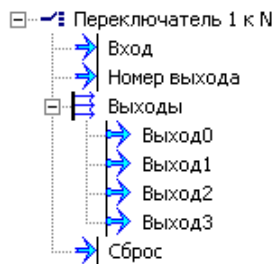
Входы и выходы:

	Обозначение	Тип
Вход	Вход	Логический
Выход	Передний фронт	Логический

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Переключатель 1 в N

На вход *Номер выхода* подается номер одного из выходов *Выход N*. На этот выход и будет подано значение входа *Вход*. Значение *Номер выхода* может быть больше, чем всего выходов. В этом случае нужный номер выхода будет рассчитываться как остаток от деления числа, которое подано на вход *Номер выхода* на число имеющихся выходов. Метка времени и признаки качества входа передаются на выход. Обнуление выходов происходит в зависимости от настройки свойства *Обнулять остальные выходы* на странице *Свойства*. Если *Обнулять остальные выходы - Вкл*, то обнуление выходов происходит по переднему фронту импульса на входе *Сброс*. Если *Обнулять остальные выходы - Выкл*, то обнуление выходов происходит независимо от значения на входе *Сброс*. Вид функционального блока в дереве объекта:



Настройки:

Свойство	Тип свойства	Значение
Число выходов	Целый	От 1 до 255
Обнулять остальные входы	Логический	

Входы и выходы:

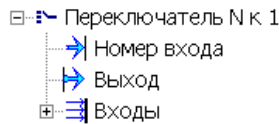
	Обозначение	Тип
Входы	Сброс	Логический
	Номер выхода	Целый
Выходы	Вход	Вещественный
	Выход 0	Вещественный

	Выход N	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Переключатель N в 1

На *Номер входа* подается число, номер одного из входов *Вход N*. Значение этого входа и подается на *Выход*. Номер входа может быть больше, чем всего входов. В этом случае нужный вход рассчитывается как остаток от деления числа, поданного на *Номер входа* на число имеющихся входов. *Метка времени и признаки качества входа передаются на выход*. Вид функционального блока в дереве объекта:



Настройки:

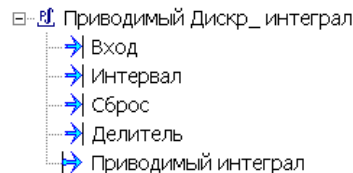
Свойство	Тип свойства	Значение
Число выходов	Целый	От 1 до 255
Входы и выходы:		
	Обозначение	Тип
Входы	Номер входа	Целый
	Вход 0	Вещественный

	Вход_N	Вещественный
Выходы	Выход	Вещественный

Поддерживаемый тип опроса – любой. Восстановление при рестарте - не требуется.

Приводимый дискретный интеграл

Аналогичен *Дискретному интегралу*, но поделенный на *Делитель*. Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*. Вид функционального блока в дереве объекта:



Дисциплина опроса должна быть периодической. Хорошая точность достигается, когда *Период опроса* по крайней мере в 20 раз меньше *Интервала*.

Входы и выходы:

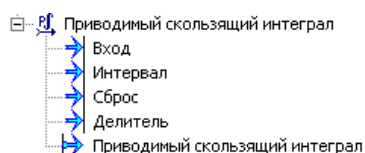
	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Делитель	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Приводимый скользящий интеграл

Аналогичен *Скользящему интегралу*, но поделенный на *Делитель*. Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*.

Вид функционального блока в дереве объекта показан на рисунке ниже.



Дисциплина опроса должна быть периодической. Хорошая точность достигается, когда *Период опроса* по крайней мере в 20 раз меньше *Интервала*.

Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Делитель	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Реальное дифференцирующее звено

Функциональный блок *Реальное дифференцирующее звено* осуществляет математическую операцию дифференцирования поданного на вход аналогового сигнала. Логика работы функционального блока заключается в следующем - реальное дифференцирующее звено представлено в виде последовательного соединения двух звеньев: дифференцирующего и апериодического звена первого порядка (фильтр). В свойствах настроек ФБ вводятся два коэффициента:

- Kd – коэффициент дифференциатора;
- Td – постоянная времени дифференцирования.

При равенстве Td и $t0$ ($t0$ – период, с которым проводится вычисление задачи) фильтр отсутствует. В этом случае получается идеальное дифференцирующее звено. После расчета на выход ФБ передается вычисленное значение входной величины.

Возможности управления:

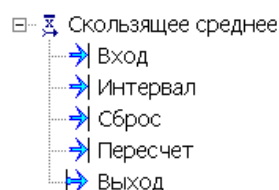
- Автоматическое, через управляющие входы;
- Ручное, через окно операторского управления.

Входы и выходы ФБ:

Имя	Тип	Примечание
Вход	Вещественный дв. точности	Вычисляемое значение
Выход	Вещественный дв. точности	Вычисленное значение

Скользящее среднее

На *Выход* подается среднее значение из пришедших на *Вход* за последние *Интервал* секунд. Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*. Вид функционального блока в дереве объекта:



Дисциплина опроса должна быть периодической. Хорошая точность достигается, когда *Период опроса* по крайней мере в 20 раз меньше *Интервала*.

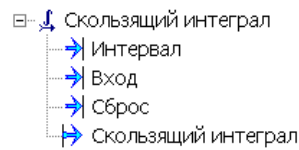
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Скользящий интеграл

На *Выход* подается интеграл значений, пришедших на *Вход* за последние *Интервал* секунд. Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*. Вид функционального блока в дереве объекта:



Дисциплина опроса должна быть периодической. Хорошая точность достигается, когда *Период опроса* по крайней мере в 20 раз меньше *Интервала*.

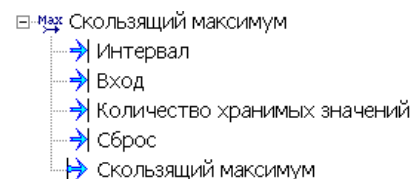
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Скользящий максимум

На *Выход* подается максимальное из значений, которые приходили на *Вход* в течение последних *Интервалов секунд*. Если за это время на вход пришло более чем *Количество хранимых значений*, самые старые из пришедших удаляются. Переход *Сброс* из *Ложь* в *Истина* дает команду начать расчет заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*. Вид функционального блока в дереве объекта:



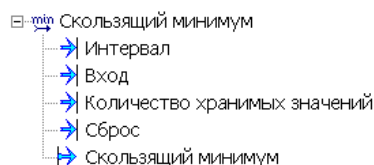
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Количество хранимых значений	Целый
	Сброс	Логический
Выходы	Скользящий максимум	Вещественный

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Скользящий минимум

На *Выход* подается минимальное из значений, которые приходили на *Вход* в течение последних *Интервал* секунд. Если за это время на вход пришло более чем *Количество хранимых значений*, самые старые из пришедших удаляются. Переход *Сброс* из *Ложь* в *Истина* дает команду начать расчет заново. Изменение *Интервала* равносильно переходу *Сброс* из *Ложь* в *Истина*. Вид функционального блока в дереве объекта:



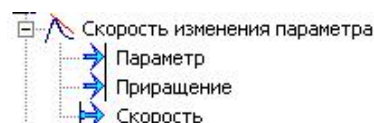
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	Интервал	Вещественный
	Количество хранимых значений	Целый
	Сброс	Логический
Выходы	Минимум	Вещественный

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Скорость изменения параметра

Производит расчет скорости изменения параметра на заданном интервале времени. Вид в дереве:



Входы и выходы:

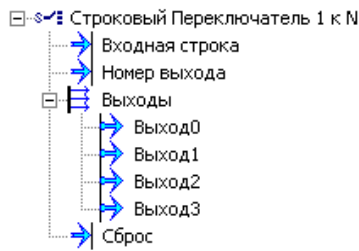
Название	Тип	Назначение
Параметр	Вход. Аналоговый	Подключается параметр, скорость изменения которого необходимо вычислять.
Приращение	Вход. Аналоговый	Задается интервал времени. В формуле расчета скорости это параметр h . Величина приращения должна быть такой, чтобы при максимальной частоте изменения фильтрованного параметра на интервале монотонности (приблизительно одна четверть периода при периодическом сигнале) укладывалось не менее $(3 - 4) \cdot h$. Задается в секундах.
Скорость	Выход. Аналоговый	Выводится скорость изменения параметра. Вычисляется по формуле: $F'(X_i) = [21f(X_i) - 13f(X_i - h) - 17f(X_i - 2h) + 9f(X_i - 3h)] / 20 \cdot h, (10)$ где: $f(X_i)$, $f(X_i - h)$, $f(X_i - 2h)$, $f(X_i - 3h)$ - значения измеряемого параметра в моменты времени $(X_i - kh)$ на выходе сглаживающего фильтра (для исключения влияния пульсации параметра).

Страница свойств *Настройки* ФБ *Скорость изменения параметра*. Максимальное приращение - определяет максимальное значение на входе *Приращение*. Если на контролируемый вход подается величина большая, чем указанная в данном свойстве, то в формуле расчета скорости используется значение, заданное в этом поле.

Переключатель 1 в N (строковый)

Работает аналогично Переключатель 1 к N, но со строками вместо вещественных значений.

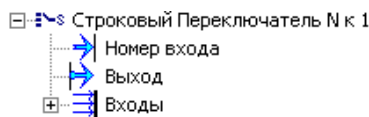
Вид функционального блока в дереве объекта представлен на картинке ниже.



Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Переключатель N в 1 (строковый)

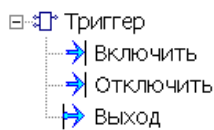
Работает аналогично Переключатель N к 1, но со строками вместо вещественных значений. Вид функционального блока в дереве объекта:



Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Триггер

При переходе входа *Включить* из *Ложь* в *Истина* на *Выход* подается *Истина*. При переходе входа *Отключить* из *Ложь* в *Истина* на *Выход* подается *Ложь*. Значение запоминается до следующего перехода одного из входов. В случае если оба входа перешли из *Ложь* в *Истина* одновременно, выдается соответствующее сообщение и новое значение *Выхода* не формируется. Вид функционального блока в дереве объекта:



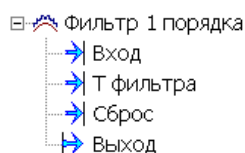
Входы и выходы:

	Обозначение	Тип
Входы	Включить	Логический
	Отключить	Логический
Выходы	Выход	Логический

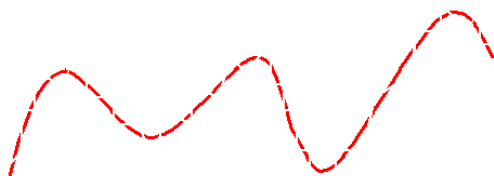
Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Фильтр первого порядка

Производит сглаживание входного сигнала. Вид функционального блока в дереве объекта:



Требуется периодическая дисциплина опроса с периодом опроса $\leq T$ фильтра. При больших значениях входной сигнал сглаживается почти до постоянной величины.



Если произошел переход *Сброс* из *Ложь* в *Истина*, то расчет начинается сначала. Изменение *T фильтра* равносильно переходу *Сброс* из *Ложь* в *Истина*.

Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный
	T фильтра	Вещественный
	Сброс	Логический
Выходы	Выход	Вещественный

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Триггер с приоритетом отключения

Реализует задачу *RS-триггера* с приоритетом команды *reset* (сброс).

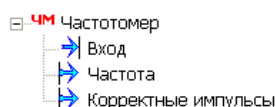
Входы и выходы:

Название	Тип	Назначение
Включить	Вход. Логический	Если вход равен значению <i>Истина</i> , то на выход проходит сигнал <i>Истина</i> , но только в том случае если вход <i>Отключить</i> имеет значение <i>Ложь</i> .
Отключить	Вход. Логический	Если вход равен значению <i>Истина</i> , то выход ФБ принимает значение <i>Ложь</i> независимо от состояния входа <i>Включить</i> . Если вход <i>Ложь</i> , то значение выхода зависит от состояния входа <i>Включить</i> : если вход <i>Включить</i> равен <i>Истина</i> , то и выход ФБ, тоже равен <i>Истина</i> , если вход <i>Включить</i> равен <i>Ложь</i> , то на выход передается предыдущее значение.
Выход	Выход. Логический	Выдает результат в зависимости от состояния входов ФБ
Инв. Выход	Выход Логический	Выдает инвертированный сигнал <i>Выхода</i> ФБ.

Инверсный выход на странице свойств *Настройки* ФБ определяет показывать или нет инверсный выход функционального блока.

Частотомер

По приходу первого счетного импульса программа включает таймер. По приходу второго импульса программа определяет ожидаемую частоту. Если третий импульс приходит с интервалом, укладывающимся в заданные границы, то программа (ФБ) выдает среднюю за два периода частоту. По приходу четвертого импульса программа усредняет частоту за три периода и т.д. Всего в буфере усреднения содержится восемь периодов. По получении девятого периода из буфера вытесняется значение первого периода. Вид функционального блока в дереве объекта:



Входы и выходы ФБ:

	Обозначение	Тип
Входы	Вход	Логический
Выход	Частота	Вещественный дв. точности
	Выход	Логический

Коэффициент понижения частоты на странице свойств *Настройки* - задается *Коэффициент понижения частоты*. *Максимальное количество периодов* - задается максимальное количество периодов в буфере усреднения

Счетчик дискретных импульсов

Считает количество переходов значения, поданного на *Вход* с *Ложь* на *Истина*. При каждом переходе значение *Число импульсов* увеличивается на единицу, пока не достигнет *Макси-*

мального значения. После этого увеличение прекращается. Счетчик работает при Сброс равном Ложь. Если Сброс принимает значение Истина, Число импульсов обнуляется.

Входы и выходы:

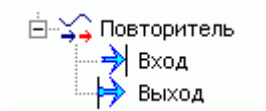
	Обозначение	Тип
Входы	Вход	Логический
	Сброс	Логический
	Макс. Значение	Целый
Выход	Число импульсов	Целый

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Повторитель

Значение, а также ОРС признак качества передается на *Выход* без изменений. Используется при построении технологической структуры объекта для удобства размещения связей, например, с ОРС переменными, чтобы входы/выходы были в одном месте.

Вид функционального блока в дереве объекта:



На закладке *Тип выхода* есть возможность выбора типа данных выхода.

По кнопке *Наследовать* наследуется тип данных *Входа*.

Входы и выходы:

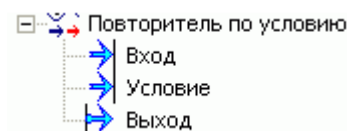
Имя	Тип
Вход	Целый
	Строковый
	Логический
	Время
	Вещественный двойной точности
Выход	Зависит от типа переменной входа

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Повторитель по условию

Значение, а также ОРС признак качества передается на *Выход* без изменений. Используется при построении технологической структуры объекта для удобства размещения связей, например, с ОРС переменными, чтобы входы/выходы были в одном месте.

Вид функционального блока в дереве объекта:



На закладке *Тип выхода* есть возможность выбора типа данных выхода. По кнопке *Наследовать* наследуется тип данных *Входа*. На закладке *Настройка* в поле *Выдавать команду на выход* устанавливается принцип передачи данных от *Выхода* к какому-либо входу (например, OPC-переменной). В случае, если на странице установлено *Ложь*, то в момент, когда ФБ начинает работу, данные от выхода, если они не изменились, передаваться не будут, т.е. будет действовать мертвая зона. Если установлено *Истина*, то данные будут передаваться от выхода принудительно.

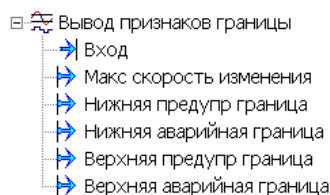
Входы и выходы:

Наименование	Тип
Вход	Вещественный двойной точности
	Время
	Логический
	Строковый
	Целый
Условие	Логический
Выход	Зависит от типа переменной входа

Режим работы - в случае, если *Условие* принимает значение *Истина*, то значение *Входа* передается на *Выход*, в противном случае функциональный блок не опрашивается. Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Вывод признаков границы

Выход принимает значение *Истина*, если нарушена соответствующая граница у значения, которое подается на *Вход*. Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Вход	Вход	Вещественный двойной точности
Выходы	Макс скорость изменения	Логический
	Нижняя предупредительная граница	Логический
	Нижняя аварийная граница	Логический
	Верхняя предупредительная граница	Логический
	Верхняя аварийная граница	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Вывод признаков качества

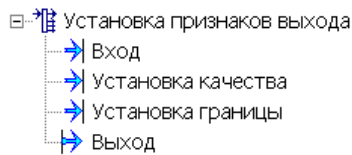
Выход принимает значение *Истина*, если величина, подаваемая на *Вход* имеет соответствующий признак качества. Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется. Входы и выходы:

	Обозначение	Тип
Вход	Вход	Вещественный двойной точности
Выходы	Ошибка конфигурации	Логический
	Нет соединения	Логический
	Ошибка устройства	Логический
	Ошибка датчика	Логический
	Последнее известное значение	Логический
	Нет связи	Логический
	Не обслуживается	Логический

Установка признаков выхода

Признаки *Входа* обнуляются, затем устанавливаются новые признаки в соответствии с значениями на входах *Установка качества* и *Установка границы* после чего данные передаются на *Выход*.

Вид функционального блока в дереве объекта:



Значения входа *Установка качества*:

Значение входа	Значение признака качества	Описание	Значение по стандарту OPC
0	Не меняется	Признаки входа остаются неизменными.	Наследуется от входа
1	Ошибка конфигурации	Алгоритм имеет проблемы с конфигурацией, например, не все связи или настройки заданы.	Opс_quality_config_error = 4
2	Нет соединения	Алгоритм имеет проблемы со связью, например, у входа алгоритма оборвана связь с выходом алгоритма-источника данных.	Opс_quality_not_connected = 8
3	Ошибка устройства	Предписанные алгоритмом операции нельзя выполнить (например, деление на нуль или переполнение порядка). Хотя бы одна входная переменная не удовлетворяет условиям, накладываемым на входные переменные алгоритмом, или хотя бы одна входная переменная пришла с любым из плохих признаков качества.	Opс_quality_device_failure = 12
4	Ошибка датчика	Сбой связи. Последнее опрошенное значение доступно. Время опроса имеется в отметке времени, сопровождающей значение.	Opс_quality_sensor_failure = 16
5	Последнее известное значение	Связи с источником данных для какого-либо из входов нет. Последнее опрошенное значение не существует. Или алгоритм работает, но выходное значение пока еще не сформировано (например, расчет интеграла).	Opс_quality_last_known = 20
6	Нет связи	При существующих настройках алгоритма данный выход не формируется.	Opс_quality_comm_failure = 24
7	Не обслуживается	Значение недостоверно	Opс_quality_out_of_service = 28
8	Значение недостоверно	Значение не определено	Opс_quality_bad = 0
9	Значение не определено	Признак качества <i>Норма</i>	Opс_quality_uncertain = 64
10	Признак качества <i>Норма</i>		Opс_quality_good = 192

При значениях входа меньше 0 и больше 9 все признаки качества устанавливаются в *Норма*.

Значения входа *Установка границы*:

Значение входа	Описание
0	Не меняется.
1	Максимальная скорость изменения нарушена.
2	Нижняя предупредительная граница нарушена.
3	Нижняя аварийная граница нарушена.
4	Верхняя предупредительная граница нарушена.
5	Верхняя аварийная граница нарушена.

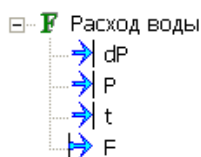
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Вещественный двойной точности
	Установка качества	Целый
	Установка границы	Целый
Выход	Выход	Вещественный двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Расход воды

Данный ФБ вычисляют расход в соответствии с ГОСТом (ГОСТ 8.563.1 (2)-97). Указанием Госстандарта России. Он введен в действие с 01.01.99.

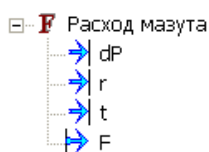


Страница свойств *Настройки* ФБ *Расход воды*.

Общие		Опрос	Настройки
☐ Уставки			
Диаметр отверстия			0
Диаметр трубопровода			0
Коэффициент притупления			0
Коэффициент СУ			0
Коэффициент трубопровода			0
Эквивалентная шероховатость			0
Диаметр отверстия			
Диаметр отверстия сужающего устройства при 20°C мм			
		Применить	Отменить
		Справка	

Расход мазута

Данный ФБ вычисляют расход в соответствии с ГОСТом (ГОСТ 8.563.1 (2)-97). Указанием Госстандарта России. Он введен в действие с 01.01.99.

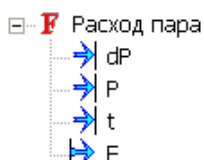


Страница свойств *Настройки* ФБ *Расход мазута*.

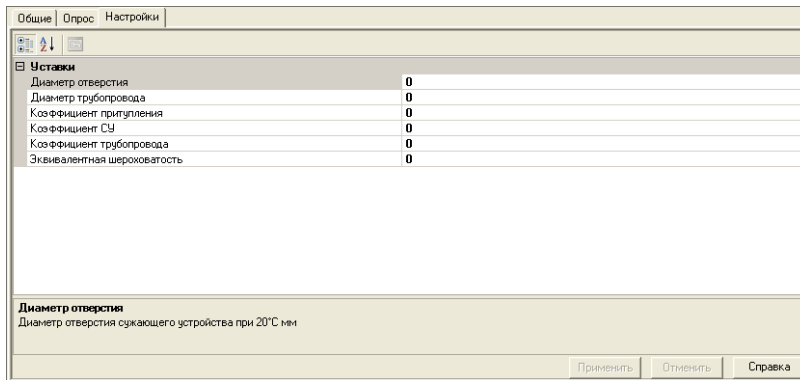
Общие		Опрос	Настройки
☐ Уставки			
Диаметр отверстия			0
Диаметр трубопровода			0
Коэффициент притупления			0
Коэффициент СУ			0
Коэффициент трубопровода			0
Эквивалентная шероховатость			0
Диаметр отверстия			
Диаметр отверстия сужающего устройства при 20°C мм			
		Применить	Отменить
		Справка	

Расход пара

Данный ФБ вычисляют расход в соответствии с ГОСТом (ГОСТ 8.563.1 (2)-97). Указанием Госстандарта России. Он введен в действие с 01.01.99



Страница свойств *Настройки* ФБ *Расход пара*

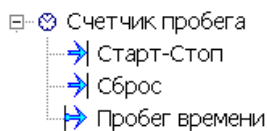


Работа со временем

Счетчик пробега

Складывает промежутки времени. Когда на *Старт* подается значение *Истина*, к значению *Выход* начинает прибавляться текущий промежуток времени в секундах. Когда на *Старте* – *Ложь* счетчик останавливается. Переход параметра *Сброс* из *Ложь* в *Истина* - обнуляет *Выход*.

Вид функционального блока в дереве объекта:



Этот функциональный блок полезно также использовать как вспомогательный для получения линейного сигнала в случае, когда требуется подавать время на какой-либо вход.

Входы и выходы:

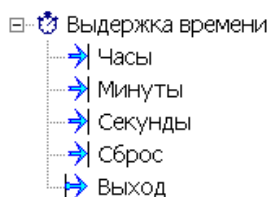
	Обозначение	Тип
Входы	Старт	Логический
	Сброс	Логический
Выход	Пробег времени	Вещественный (сек)

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - поддерживается.

Выдержка времени

При переходе *Сброс* из *Ложь* в *Истина* на *Выход* подается *Истина* и начинается отсчет установленного времени *Часы* + *Минуты* + *Секунды*. По истечении этого времени на *Выход* опять подается *Ложь* и снова ожидается переход *Сброс* из *Ложь* в *Истина*. Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Вид функционального блока в дереве объекта:



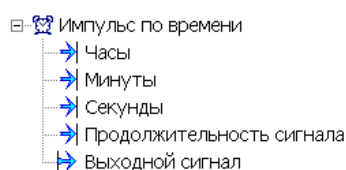
Входы и выходы:

	Обозначение	Тип
Входы	Часы	Целый
	Минуты	Целый
	Секунды	Целый
	Сброс	Логический
Выходы	Выход	Логический

Импульс по времени

В момент наступления астрономического времени *Часы* : *Минуты* : *Секунды* на *Выход* подается *Истина* в течение промежутка времени Продолжительность сигнала, после чего на *Выход* опять подается *Ложь*.

Вид функционального блока в дереве объекта:



Входы и выходы:

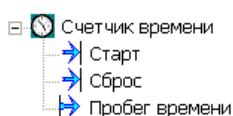
	Обозначение	Тип
Входы	Часы	Целый
	Минуты	Целый
	Секунды	Целый
	Продолжительность сигнала	Вещественный (сек)
Выходы	Выход	Логический

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Счетчик времени

Функциональный блок осуществляет счет времени присутствия логической единицы на *Входе* *Старт*. При поступлении на *Вход* *Сброс* логической единицы значение *Пробега времени* обнуляется.

Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Входы	Старт	Логический
	Сброс	Логический
Выход	Пробег времени	Вещественный дв. точности

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется. На странице свойств *Настройки* в свойстве *Единица времени* задается единица измерения *Пробега времени* (Миллисекунды, Секунды, Минуты, Часы).

Переключатель по времени

Передаёт на единственный *Выход* значение того *Входа*, который имеет самую *Свежую* метку времени.

Свойства:

Название	Тип свойства	Примечание
Количество Входов	целый	от1 до 256

Входы и выходы:

Название	Тип	Примечание
Входы ВходN	Вещественный	Группа входов. Количество устанавливается на странице <i>Свойства</i> у ФБ.
Выходы Выход	Вещественный	Выход

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Задержка по времени

Этот ФБ позволяет отсекал импульсы меньшей длительности, чем задано. Выдает на *Выход* ФБ передний фронт с *Входа* через заданный на входах *Часы*, *Минуты*, *Секунды* промежуток времени. Задний фронт на выход передается без задержки, сразу же при срабатывании его на входе. Если на *Входе* ФБ сработал передний фронт и до того, как время задержки истекло, на *Вход* ФБ подается задний фронт, то передний фронт на *Выход* ФБ через заданный промежуток времени уже не передается. Таким образом, сигнал *Вкл.* с *Входа* передается на *Выход* только в том случае, если он присутствует на *Входе*, не изменяясь в течение заданного времени.

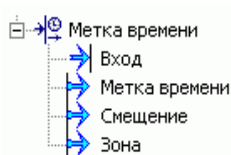
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Логический
	Часы	Целый
	Минуты	Целый
	Секунды	Целый
Выход	Выход	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Метка времени

Выводит на *Выход* метку времени значения, подаваемого на *Вход*. Вид функционального блока в дереве объекта:



Входы и выходы:

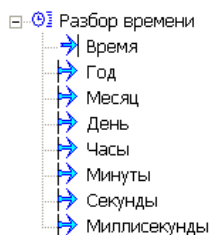
	Обозначение	Тип
Вход	Вход	Вещественный двойной точности
Выход	Метка Времени	Дата

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Разбор времени

При подаче на *Вход* *Время* текущего значения времени на *Выходах* функционального блока появляются соответствующие значения текущего *Года*, *Месяца*, *Дня*, *Часа*, *Минуты*, *Секунды* и *Миллисекунды* (функциональный блок *Разбирает* время на составляющие).

Вид функционального блока в дереве объектов представлен на картинке ниже.



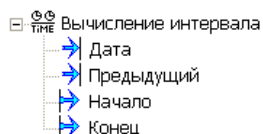
Входы и выходы:

	Обозначение	Тип
Входы	Время	Время
	Год	Целый
	Месяц	Целый
	День	Целый
Выходы	Часы	Целый
	Минуты	Целый
	Секунды	Целый
	Миллисекунды	Целый

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Вычисление интервала

Определяет интервалы времени текущий и предыдущий (декада, смена). Вид функционального блока в дереве объекта:



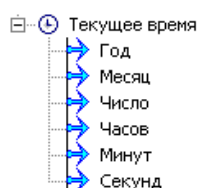
Входы и выходы:

	Обозначение	Тип	Описание
Входы	Дата	Время	Текущее время (дата) При логическом нуле (<i>Ложь</i>) на <i>Выходах</i> определен текущий интервал времени, при логической единице (<i>Истина</i>) - определен предыдущий интервал.
	Предыдущий	Логический	
Выходы	Начала	Время	В режиме исполнения показано начало интервала времени (декады, смены)
	Конец	Время	В режиме исполнения показан конец интервала времени (декады, смены)

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется. На странице свойств *Настройки* ФБ *Вычисление интервала* можно задать тип преобразования - выбирается тип преобразования интервала времени (декада или смена).

Текущее время

Выходы соответствуют текущей дате и текущему времени. Вид функционального блока в дереве объекта:



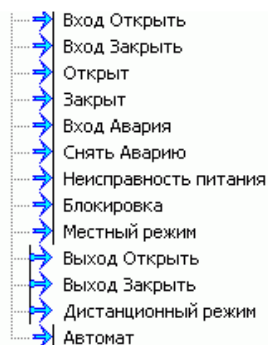
Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Исполнительные механизмы

2-х позиционный механизм с выбором изображения

Визуальный функциональный блок *Двухпозиционный механизм* с выбором изображения предназначен для управления *Насосом*, *Задвижкой* и подобными механизмами.

Вид функционального блока в дереве объекта:



В зависимости от значений входов *Открыт*, *Закрыт* и *Авария*, можно выделить 8 состояний механизма (также учитывается предыдущее состояние). Для каждого состояния выбирается свое изображение из файла. Состояния исполнительного механизма (ИМ) в зависимости от значений входов:

Состояние	Вход Открыт	Вход Закрыт	Вход Авария	Предыдущее состояние
Открыто	Вкл.	Выкл.	Выкл.	Любое
Открывается	Одинаковые		Выкл.	Закрыто Закрыто <i>Аварийное</i> Открывается Открывается <i>Аварийное</i>
Закрывается	Одинаковые		Выкл.	Открыто Открыто <i>Аварийное</i> Закрывается Закрывается <i>Аварийное</i>
Закрыто	Выкл.	Вкл.	Выкл.	Любое
Открыто <i>Аварийное</i>	Вкл.	Выкл.	Вкл.	Любое
Открывается <i>Аварийное</i>	Одинаковые		Вкл.	Закрыто Закрыто <i>Аварийное</i> Открывается Открывается <i>Аварийное</i>
Закрывается <i>Аварийное</i>	Одинаковые		Вкл.	Открыто Открыто <i>Аварийное</i> Закрывается Закрывается <i>Аварийное</i>
Закрыто <i>Аварийное</i>	Выкл.	Вкл.	Вкл.	Любое

Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Ниже представлен полный перечень входов и выходов ИМ. В зависимости от настроек некоторые из входов или выходов могут отсутствовать в связи с ненужностью. Найдите и внимательно прочитайте статью о настройках ИМ.

	Имя	Тип	Примечание
Входы	Открыт (Включен)	Логический	КВО
	Открыть (Включить)	Логический	Команда, в <i>Автоматическом режиме</i> передается на выход
	Закрыто (Отключен)	Логический	КВЗ
	Закрыть (Отключить)	Логический	Команда, в <i>Автоматическом режиме</i> передается на выход
	Авария	Логический	При настройке формирования аварии от входа, его значение передается на <i>Выход</i> и состояние аварии зависит от него.
	Снять аварию	Логический	Снимает только мигание сигнальной области, состояние аварии не снимается.
	Неисправность питания	Логический	Дополнительная информация. Если <i>Истина</i> , появляется такая надпись в окне оператора
Блокировка	Логический	Включает режим <i>Блокировки</i> . Изображение ИМ в мнемосхеме при этом обводится в черную рамку.	

	Имя	Тип	Примечание
	Местный режим	Логический	Включает режим <i>Местного управления</i> . Изображение ИМ в мнемосхеме при этом становится бледным.
	Автомат	Логический	В зависимости от настройки <i>Запрет изменения режима в окне управления</i> - если флаг стоит, состояние режима Автоматический/Дистанционный осуществляется по текущему значению этого входа если флаг не стоит, по переднему фронту производится однократное переключение режима в Автоматический.
Выходы	Открыть (Включить)	Логический	Команда
	Заккрыть (Отключить)	Логический	Команда
	Дистанционный режим	Логический	Команда

Режимы работы ФБ

- *Местный режим*. Этот режим используется в случае, когда управление исполнительным механизмом осуществляется извне, не зависимо от ВФБ. Включается, если вход Местный режим равен *Истина*. При этом выходам *Открыть* и *Заккрыть* присваивается значение *Ложь*, в окне оператора отображается включение *Местного режима* и для оператора становятся недоступными все команды, кроме квитирования аварии;
- *Блокировка*. Этот режим используется в случае, когда управление исполнительным механизмом осуществляется извне, не зависимо от ВФБ. Включается, если вход Блокировка равен *Истина*. При этом выходам *Открыть* и *Заккрыть* присваиваются значения в зависимости от настройки *Действия при Блокировке*, в окне оператора отображается включение режима *Блокировки* и для оператора становятся недоступными все команды, кроме квитирования аварии. Изображение ИМ в мнемосхеме при этом обводится в черную рамку;
- *Автоматический режим*. Этот режим используется, когда команды формируются вне ВФБ, но управление ИМ осуществляется через функциональный блок. При этом выходам *Открыть* и *Заккрыть* присваиваются значения входов *Открыть* и *Заккрыть* (сформированных где-то вне блока), а команды в окне оператора становятся недоступными. Выходу *Дистанционный режим включен* присваивается *Ложь*;
- *Дистанционный режим*. Используется для управления ИМ из окна оператора. При переходе в *Дистанционный режим* у выходов *Открыть* и *Заккрыть* остаются предыдущие значения, в дальнейшем значения этих выходов формируются при нажатии соответствующих кнопок в окне оператора. Выходу *Дистанционный режим включен* присваивается *Истина*.

Переключение режимов *Автоматический/Дистанционный* зависит от настройки *Запрет переключения режимов в окне управления*. Если запрет включен (стоит галочка), то режим переключается в зависимости от входа *Автомат*, если запрет отключен, то переключается кнопками в окне оператора (по переднему фронту входа *Автомат* тоже может произойти переключение в *Автоматический режим*, кнопки при этом остаются доступными для ручного переключения режима).

Состояние *Авария* сопровождается следующими признаками:

- Характерным изображением в мнемосхеме;
- Аварийным сообщением (если оно не отключено);
- Выходу *Авария* (если он есть) присваивается значение *Истина*.

Функциональный блок имеет страницу свойств *Настройки*, на которой позволяет выбрать два способа формирования состояния *Авария*:

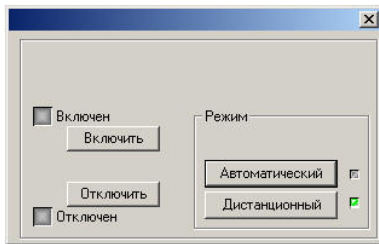
- *От входа Авария*. Используйте этот способ если сигнал *Авария* формируется вне функционального блока. В этом случае состояние *Авария* будет формироваться сразу в зависимости от значения этого входа и передаваться на выход *Авария* (если выход есть);
- *По несрабатыванию* (следует указать максимальное время срабатывания). В этом случае (независимо от режима работы ВФБ, *Дистанционного* или *Автоматического*) если изменился один из выходов *Открыть* или *Заккрыть* включается счетчик времени. Пока истекшее время не превышает максимальное время срабатывания (введенное на странице с настройками) выходу *Авария* присваивается значение "*Ложь*", затем проверяются следующие условия:
 - Вход *Открыт* (Включен) равен выходу *Открыть* (Включить);
 - Вход *Заккрыт* (Отключен) равен выходу *Заккрыть* (Отключить);

- Вход *Открыт* (Включен) не равен выходу *Закрыт* (Отключен).

Входы, определяемые состоянием механизма должны соответствовать команде. Если используются не все из перечисленных входов или выходов, значение отсутствующего входа или выхода считается инверсным по отношению к тому, который есть. *Авария* формируется при *несоблюдении хотя бы одного из трех условий*. Если не используются выходы *Открыть* (Включить) и(или) *Закрыть* (Отключить), то авария по несрабатыванию не формируется. Если выбраны оба способа формирования, то достаточно выполнения хотя бы одного. Для выбора нужного способа (ов) используйте поле признаков на странице свойств *Настройки*. Функциональный блок имеет следующие сообщения:

Сообщение	Условие	Категория	По умолчанию
Сигнал <i>Открыть</i>	Оператором нажата кнопка <i>Открыть</i>	Информация	нет
Сигнал <i>Закрыть</i>	Оператором нажата кнопка <i>Закрыть</i>	Информация	нет
Сигнал <i>Стоп</i> (только для задвижек)	Оператором нажата кнопка <i>Закрыть</i>	Информация	нет
Перевод в <i>Автоматический режим</i>	Оператором нажата кнопка <i>Автоматический</i>	Информация	нет
Перевод в <i>Дистанционный режим</i>	Оператором нажата кнопка <i>Дистанционный</i>	Информация	нет
Авария квитирована	Оператором нажата кнопка <i>Квитировать</i>	Информация	нет
Авария	Выход <i>Авария</i> перешел из <i>Ложь</i> в <i>Истина</i>	Авария	есть

На странице *Сообщения* в свойствах функционального блока вы можете настроить вывод этих сообщений установкой и снятием соответствующих флажков. *Окно оператора* появляется на экране при щелчке мышью на исполнительном механизме (только в режиме исполнения). С помощью этого окна оператор может изменять состояние выходов ИМ, квитировать аварию, переключать режимы управления.



Функции элементов управления описаны ниже:

Элемент	Назначение
Надпись <i>Авария</i>	Появляется, если сформированный (независимо от режима) выход <i>Авария</i> равен <i>Истина</i> . Иначе надпись отсутствует.
Кнопка <i>Режим</i>	Появляется совместно с надписью <i>Авария</i> . При нажатии на эту кнопку снимается мигание сигнальной области и формируется соответствующее сообщение.
Надпись <i>Неисправность питания</i>	Информация для оператора, появляется если соответствующий вход равен <i>Истина</i> .
Надпись <i>Местный</i>	Появляется, если соответствующий вход равен <i>Истина</i> . При этом кнопки <i>Открыть</i> , <i>Закрыть</i> , <i>Автоматический</i> , <i>Дистанционный</i> становятся недоступными.
Кнопка <i>Автоматический</i>	Включает <i>Автоматический режим</i> . При этом кнопки <i>Открыть</i> и <i>Закрыть</i> становятся недоступными.
Кнопка <i>Дистанционный</i>	Включает <i>Дистанционный режим</i> . Становятся доступными все кнопки.
Индикаторы <i>Открыто</i> и <i>Закрыто</i>	Сигнализируют о наличии сигналов на входах <i>Открыт</i> и <i>Закрыт</i> соответственно.
Индикаторы <i>Открыть</i> и <i>Закрыть</i>	Сигнализируют о наличии сигналов на выходах <i>Открыть</i> и <i>Закрыть</i> соответственно. Если установлена настройка <i>Импульсные выходы</i> , лампочка становится серой после того как управляющий импульс прошел.
Кнопка <i>Открыть</i>	Выходу <i>Открыть</i> присваивается <i>Истина</i> , выходу <i>Закрыть</i> - <i>Ложь</i> .
Кнопка <i>Стоп</i> (только у Задвижки)	Выходу <i>Открыть</i> присваивается <i>Ложь</i> , выходу <i>Закрыть</i> - <i>Ложь</i> , но авария по несрабатыванию при этом не формируется (пока не будет нажата какая-то другая кнопка).
Кнопка <i>Закрыть</i>	Выходу <i>Открыть</i> присваивается <i>Ложь</i> , выходу <i>Закрыть</i> - <i>Истина</i> .
Надпись <i>Блокировка</i>	Появляется, если соответствующий вход равен <i>Истина</i> . При этом кнопки <i>Открыть</i> , <i>Закрыть</i> , <i>Автоматический</i> , <i>Дистанционный</i> становятся недоступными.
Список <i>Недостовверные входы</i>	Этот список появляется только в том случае, если данные хотя бы одного из входов функционального блока имеют <i>Плохой</i> признак качества (изображение ИМ в мнемосхеме при этом бледнеет). Кнопка <i>Квитировать</i> появляется совместно со списком <i>Недостовверные входы</i> . При нажатии на эту кнопку изображение ИМ в мнемосхеме перестает быть бледным вплоть до появления новых признаков недовостовренности.

Вид страницы свойств:

Группа *Выходы команд*:

- *Оба* - в дереве объектов отображаются оба выхода, *Открыт* и *Закреть*. При этом, если используется только один вход состояния, то в *Автоматическом режиме* на один из выходов будет подаваться соответствующий ему сигнал от входа, на другой выход - инверсия этого сигнала. (При двух входах состояния передаются оба сигнала без изменений);
- *Открыт* (Включен) отображается только этот выход;
- *Закреть* (Отключен) отображается только этот выход;
- *Нет* - управление отсутствует. В этом случае не будут отображаться:
 - Входы *Включить* и *Отключить*;
 - Вход *Местный режим*;
 - Выходы *Включить* и *Отключить*;
 - Выход *Авария*;
 - Выход *Дистанционный режим*;

Настройки, связанные с поведением этих элементов тоже отображаться не будут, и в режиме исполнения не будет открываться окно управления.

Группа *Выходы состояния*:

- *Оба* - в дереве объектов отображаются оба (*Открыт* и *Закреть*) входа ФБ. В этом случае возможны переходные состояния (открывается, закрывается), если оба входа имеют одинаковое значение;
- *Открыт* (Включен) - отображается только вход *Открыт*. Вход *Закреть* не отображается в дереве и всегда считается равным *Не открыт*;
- *Закреть* (Отключен) – аналогично;
- *Нет* - не отображается ни один вход. Очевидно, при такой настройке ИМ не может использоваться для отображения состояния, а только для управления. Поэтому изображение ВФБ в мнемосхеме в этом случае демонстрирует состояние не входов, а команд на выходе, (можно считать для ясности, что *Входы* состояния равны *Выходам* команд, в отношении же *Блокировки* и *Аварии* - как обычно).

Группа *Входы внешних команд* (входы существуют для автоматического режима работы, их значения передаются на соответствующие выходы ФБ, если используется только один из входов внешней команды (не оба), то второй формируется инверсно):

- *Оба* - в дереве объектов отображаются оба входа;
- *Открыт* (Включен) - только этот вход;
- *Закреть* (Отключен) - только этот вход;
- *Нет* - ни одного из этих входов. В этом случае управление не может быть автоматическим, поэтому не используются *Вход Автомат* и *Выход Дистанционный режим*, переключение режима в окне управления не доступно.

Состояние *Авария формировать*:

- *От входа Авария* (включает/отключает формирование аварии от входа *Авария* и отображает/прячет этот вход);
- *По несрабатыванию* (доступно только при наличии управления).

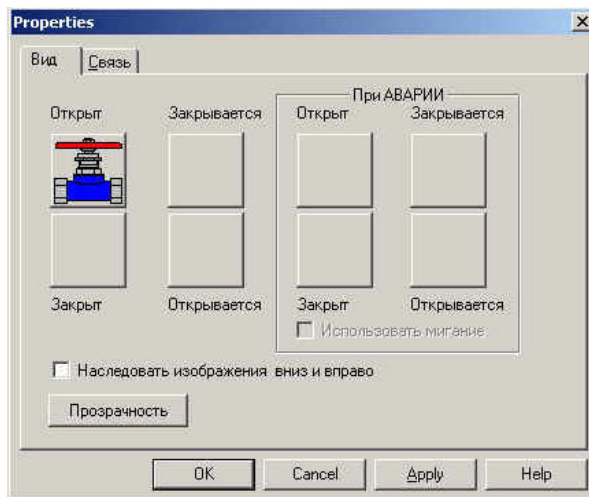
Действия при блокировке:

- *Не переключать* - не изменять текущее состояние *Выходов* команд пока *Блокировка* равна *Истина*;
- *Открыть* - установить *Выходы Открыть* (Включить) в *Истина*, *Заккрыть* (Отключить) - в *Ложь* и не изменять пока *Блокировка* равна *Истина*;
- *Заккрыть* - аналогично *Открыть*, только наоборот.

Если установлена галочка *Импульсные*, то выходы команд изменяются импульсно. Настройка становится доступной только если используются оба выхода команд. Продолжительность импульса задается в окне *Продолжительность импульса* в секундах. При режиме аварии По *несрабатыванию* запоминаются значения, которые были на момент формирования импульса, то есть окончание импульса на алгоритм не влияет. Запрет на изменение режима... - установка этого флажка делает недоступными кнопки переключения режима в окне управления. Режим будет переключаться только от входа *Автомат*.

- ❖ **Важно!** Если часто используется однотипный функциональный блок, поместите его в библиотеку, где он будет сохранен вместе со всеми настроенными свойствами. Таким образом, возможно избежать многократного выполнения одних и тех же настроек.

В зависимости от состояния ИМ в режиме исполнения будет отображаться один из 8-ми выбранных изображений. Чтобы выбрать изображение, откройте страницу свойств ИМ.



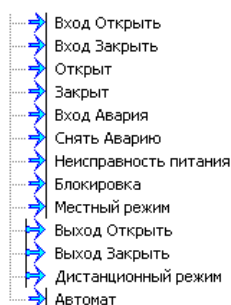
Нажатие на одну из квадратных кнопок приведет к открытию стандартного диалога для выбора файла. После этого изображение будет помещено на кнопку для предпросмотра.

В изображении возможно назначить один из цветов прозрачным, Для этого нажмите кнопку *Прозрачность*. Настройки прозрачности будут применены ко всем восьми изображениям одинаково.

Воздуходувка, Задвижка, Насос, Клапан переключения потока.

Задвижка, Воздуходувка, Насос, Клапан переключения потока имеют одинаковую функциональность и различаются названиями входов и выходов, а также представлением в мнемосхеме. Соответствующие входы у *Задвижки* называются - *Открыт, Закрыт*, у *Насоса* и *Воздуходувки, Клапана переключения потоков* и *Динамического ИМ* - *Включен, Отключен*.

Вид функционального блока в дереве объекта:

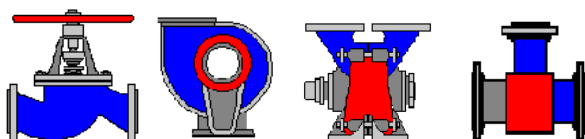


Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

- ❖ **Важно!** При использовании в качестве элемента мнемосхемы только элемента управления из комплекта ВФБ *Задвижка, Насос или Воздуходувка*, не имеющего связи с ФБ, свойство элемента *Выход Авария* приравнивается значению свойства *Вход Авария*.

Входы и выходы ВФБ аналогичны входам и выходам ВФБ 2-позиционному механизму с выбором изображений. Режимы работы, Формирование аварий, Сообщения, Управление из окна оператора, Страница свойств ФБ также аналогичны ВФБ 2-позиционному механизму с выбором изображений.

Изображение имеет две закрашиваемых области - основную и сигнальную. Цвета, которыми закрашиваются соответствующие элементы в режиме исполнения зависят от значений входов. В режиме разработки сигнальная область отображается красным цветом (у Клапана переключения потока в открытом состоянии закрашивается одна часть, в закрытом - другая). В режиме разработки выводится изображение, соответствующее состоянию *Открыт + Авария*. На рисунке ниже показаны изображения ФБ слева направо: *Задвижка, Воздуходувка, Насос, Клапан переключения потока*.



Основной цвет формируется в зависимости от значений входов *Открыт* и *Закрыт*

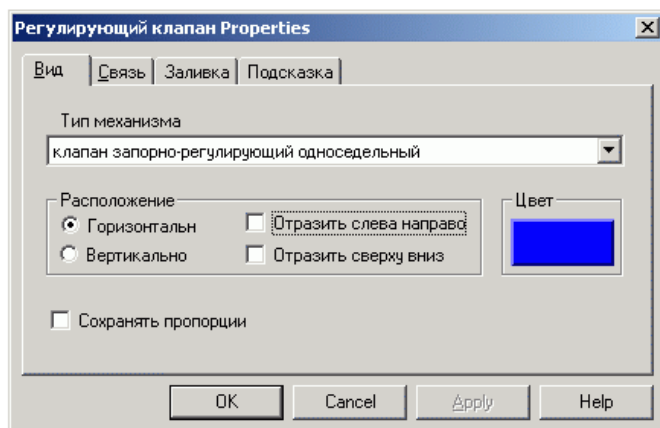
	Имя	Значение			
Входы	Открыт	Истина	Истина	Ложь	Ложь
	Закрыт	Истина	Ложь	Истина	Ложь
Основной цвет (цвет корпуса)		Настраивается	Настраивается	Серый	Серый

Кроме того, на странице свойств *Вид* функционального блока, возможно настроить такие свойства как *Тип механизма* (вариант изображения), *Ориентация в мнемосхеме*, *Основной цвет*.

Сигнальный цвет формируется в зависимости от значения, присваиваемого выходу *Авария*, и от значений входов *Открыт* и *Закрыт*

		Значение выхода Авария - Истина	Значение выхода Авария - Ложь
Входы	Если вход <i>Открыт</i> не равен входу <i>Закрыт</i>	Красный (если <i>Авария</i> не была квитиrowана - мигающий красный)	Серый
	Если вход <i>Открыт</i> равен входу <i>Закрыт</i>	Красный (если <i>Авария</i> не была квитиrowана - мигающий красный)	Желтый

Для настройки вида изображения ВФБ существует закладка *Вид*.



В зависимости от выбранного типа в списке *Тип механизма* отображается тот или иной рисунок.

В зависимости от настроек *Расположение* можно добиться любого из восьми положений рисунка:

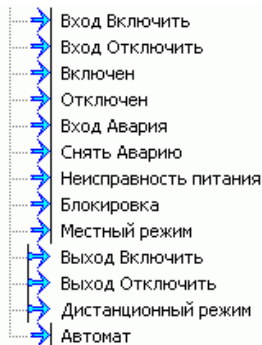
- *Горизонтально* - нормальное расположение рисунка;
- *Вертикально* - повернуть на 90 градусов;
- *Отразить слева направо* - зеркально отразить по горизонтали;
- *Отразить сверху вниз* - зеркально отразить по вертикали.

Цвет - устанавливает основной цвет механизма (цвет корпуса). Как правило, его выбирают в соответствии с цветом технологической среды (трубопровода). При нажатии на цветную кнопку всплывает стандартный диалог *Windows* для выбора цвета. Если стоит флаг *Сохранять пропорции*, то пропорции рисунка сохраняются.

Динамический исполнительный механизм

Визуальный функциональный блок *Динамический исполнительный механизм* аналогичен другим ВФБ и служит для отображения работы исполнительного механизма.

Вид функционального блока в дереве объекта:



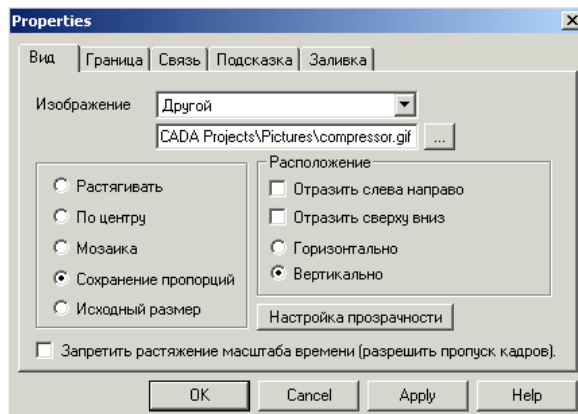
Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Входы и выходы ВФБ аналогичны входам и выходам ВФБ *2-позиционному механизму с выбором изображений*. *Режимы работы, Формирование аварий, Сообщения, Управление из окна оператора, Страница свойств ФБ* также аналогичны ВФБ *2-позиционному механизму с выбором изображений*.

Динамический ИМ в режиме исполнения отображает мультфильм (имеются встроенные изображения типичных ИМ, есть возможность выбора внешнего мультфильма). Режимы работы мультфильма зависят от состояния ФБ. Если *Динамический ИМ* включен (на *Выходе ФБ Включен - Истина*) мультфильм проигрывается, если на выходе ФБ *Отключен - Истина*, мультфильм не проигрывается. При аварии (на *Входе ФБ Авария - Истина*) все изображения мультфильма мигает.



На закладке *Вид* настраивается внешний вид ВФБ *Динамический ИМ*.



Поле ввода *Изображение* - указывает имя динамического файла (можно выбрать любой файл, нажав на кнопку обзора). Флаг *Растягивать* - задает растягивание изображения по рамке ФБ. Флаг *По центру* - задает центрированное расположение изображения.

В зависимости от настроек группы *Расположение* можно добиться любого из восьми положений рисунка:

- *Горизонтально* - нормальное расположение рисунка;
- *Вертикально* - повернуть на 90 градусов;
- *Отразить слева направо* - зеркально отразить по горизонтали;
- *Отразить сверху вниз* - зеркально отразить по вертикали.

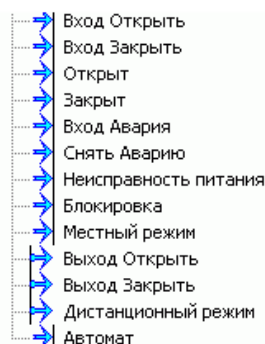
Кнопка *Настройка прозрачности* вызывает диалог для выбора цвета, который не будет выводиться при отображении картинки. Флаг *Запретить растяжение масштаба времени* (разрешить пропуск кадров) разрешает пропуск кадров динамического изображения.

- ❖ **Важно!** При отсутствии связи у входа ВФБ или работы графического редактора вне среды пакета программ MasterSCADA наследование настроек свойств параметра не работает. В этом случае все свойства индикатора настраиваются непосредственно на страницах свойств. После установления в дереве связи входа ВФБ для импорта всех необходимых настроек следует выполнить команду меню Сервис - Обновить связи.

Задвижка мнемоническая, Клапан переключения потока мнемонический

Клапан переключения потока мнемонический полностью аналогичен задвижке мнемонической. Визуальный функциональный блок Задвижка мнемоническая (КПП мнемонический) предназначен для управления задвижкой и подобными механизмами.

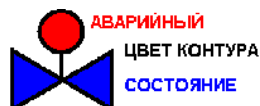
Вид функционального блока в дереве объекта:



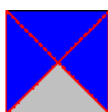
Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Входы и выходы ВФБ аналогичны входам и выходам ВФБ 2-позиционному механизму с выбором изображений. Режимы работы, Формирование аварий, Сообщения, Управление из окна оператора, Страница свойств ФБ также аналогичны ВФБ 2-позиционному механизму с выбором изображений.

В мнемосхеме ВФБ Задвижка мнемоническая имеет следующее изображение:



ВФБ Клапан переключения потока (КПП) мнемонический представлен на рисунке ниже (у КПП мнемонического в режиме исполнения в открытом состоянии закрасивается одна часть, в закрытом - другая).



У изображения механизма в мнемосхеме есть свойства, которые можно настроить. Для этого выберите пункт *Свойства*; в контекстном меню, всплывающем при нажатии правой клавиши мыши на изображении.

Группа *Расположение* содержит кнопки, с помощью которых можно получить одно из четырех положений механизма. Возможно также выбрать цвета:

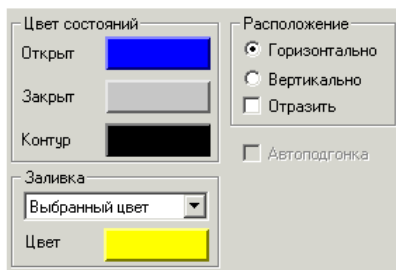
- Для открытого состояния;
- Для закрытого состояния;
- Цвет контура.

Установка поля переключателей *Автоподгонка* заставит рамку, в которой рисуется изображение автоматически принимать пропорции и размер самого изображения (эта функция доступна не во всех системах).

Цвета исполнительного механизма и их комбинации сигнализируют о его состоянии следующим образом:

Место	Цвет	Условие
Аварийный	красный-серый (мигает)	Выход <i>Авария</i> равен <i>Истина</i> и авария еще не квитирована
	красный	Выход <i>Авария</i> равен <i>Истина</i> , авария квитирована
	серый	Выход <i>Авария</i> равен <i>Ложь</i>
Цвет контура	выбранный	Механизм открыт или закрыт, т.е. выполнены одновременно 3 условия: Вход <i>Открыт</i> равен выходу <i>Открыть</i> , Вход <i>Закрыт</i> равен выходу <i>Закрыть</i> , Вход <i>Открыт</i> не равен входу <i>Закрыт</i>
	выбранный серый (мигает)	Механизм в переходном состоянии, т.е. хотя бы одно из условий (указанных выше) не выполнено
Состояние	выбранный для состояния <i>Открыт</i>	Вход <i>Открыт</i> равен <i>Истина</i>
	выбранный для состояния <i>Закрыт</i>	Вход <i>Закрыт</i> равен <i>Истина</i>
		Если значения обоих входов одинаковые, цвет не изменяется, т.е. остается как в предыдущем состоянии (но мигает контур).

Настройка изображения в мнемосхеме:



Группа *Цвета* содержит цветные кнопки для выбора цветов. Группа *Расположение* позволяет настроить расположение элемента на мнемосхеме:

- *Горизонтально* - по умолчанию;
- *Вертикально* - повернуть на 90 градусов;
- *Отразить* - повернуть на 180 градусов.

Группа *Заливка фона*:

- *Общий цвет фона* - свободное пространство заливается цветом, выбранным для фона мнемосхемы;
- *Выбранный цвет* - цвет заливки свободного пространства выбирается кнопкой (кнопка доступна только для этого способа);
- *Прозрачная* - свободное пространство не заливается.

Насос мнемонический

Функциональность ВФБ *Насос мнемонический* полностью аналогична *Задвижке мнемонической*. Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

- ❖ **Важно!** Для *Насоса* термины *Открыт*, *Закрыт* эквивалентны *Включен*, *Отключен*.

В мнемосхеме функциональный блок имеет следующее изображение:

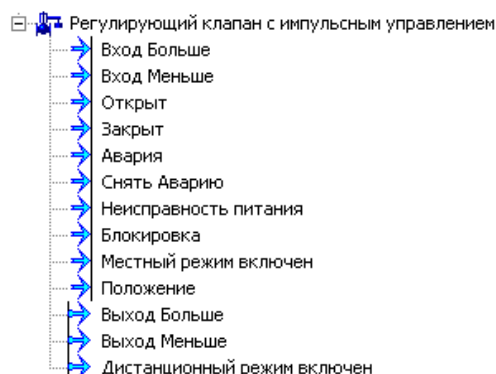


Свойства изображения на мнемосхеме настраиваются аналогично *Задвижке мнемонической*.

Регулирующий клапан с импульсным управлением

Визуальный функциональный блок *Регулирующий клапан с импульсным управлением* служит для управления регулирующим клапаном данного типа.

Вид функционального блока в дереве объекта:



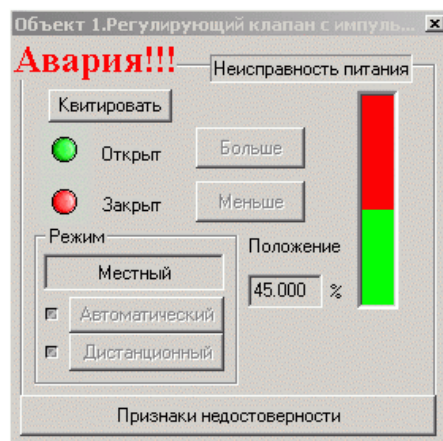
Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Ниже представлен полный перечень входов и выходов ВФБ. В зависимости от настроек некоторые из входов или выходов могут отсутствовать в связи с ненужностью.

	Имя	Тип	Примечание
Входы	Положение	Вещественный дв. точности	Определяет высоту столбика параметра в окне оператора. Должен быть заранее масштабирован от 0 до 100.
	Открыт	Логический	КВО (конечный выключатель открытого положения).
	Больше	Логический	Команда, в <i>Автоматическом режиме</i> передается на выход <i>Больше</i> .
	Закрыт	Логический	КВЗ (конечный выключатель закрытого положения).
	Меньше	Логический	Команда, в <i>Автоматическом режиме</i> передается на выход <i>Меньше</i> .
	Авария	Логический	Переключает цвет сигнальной области изображения
	Снять аварию	Логический	Снимает мигание сигнальной области при аварии (квитирование)
	Неисправность питания	Логический	Если <i>Истина</i> , появляется надпись в окне оператора.
Выходы	Блокировка	Логический	Включает режим <i>Блокировка</i>
	Местный режим включён	Логический	Включает режим <i>Местный</i>
	Больше	Логический	Команда
	Меньше	Логический	Команда
	Дистанционный режим включен	Логический	Команда

Режимы работы и Сообщения, аналогичны ВФБ 2-позиционному механизму с выбором изображений.

Окно оператора содержит необходимый набор кнопок и других элементов, с помощью которых оператор может осуществлять управление ИМ и отслеживать его состояние. Окно оператора открывается в режиме исполнения при нажатии левой клавиши мыши на изображении, т.е. доступно только из мнемосхемы.



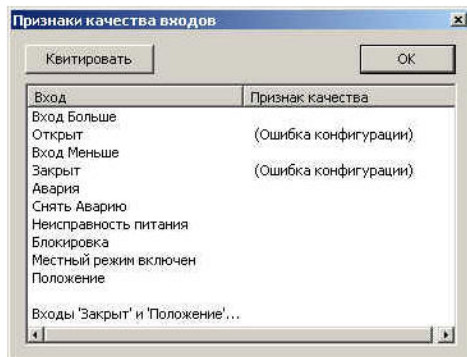
Функции элементов управления описаны в аблице ниже.

Элемент	Назначение
Надпись <i>Авария</i>	Появляется вне зависимости от режима при наличии сигнала <i>Истина</i> на входе <i>Авария</i> .
Кнопка <i>Квитировать</i>	Появляется совместно с надписью <i>Авария</i> . При нажатии на эту кнопку снимается мигание сигнальной области.
Надпись <i>Неисправность питания</i>	Информация для оператора, появляется только, если соответствующий вход равен <i>Истина</i> .
Надпись <i>Местный</i>	Появляется если соответствующий вход равен <i>Истина</i> , при этом кнопки <i>Открыть</i> , <i>Закрыть</i> , <i>Автоматический</i> , <i>Дистанционный</i> становятся недоступными.
Кнопка <i>Автоматический</i>	Включает <i>Автоматический режим</i> , при этом кнопки <i>Открыть</i> и <i>Закрыть</i> становятся недоступными.
Кнопка <i>Дистанционный</i>	Включает <i>Дистанционный режим</i> . Становятся доступными все кнопки.
Индикатор <i>Открыт</i>	Если вход <i>Открыт</i> равен <i>Истина</i> индикатор становится зеленым, иначе - серым.
Индикатор <i>Закрыт</i>	Если вход <i>Закрыт</i> равен <i>Истина</i> индикатор становится красным, иначе - серым.
Кнопка <i>Больше</i>	При нажатии выходу <i>Больше</i> присваивается <i>Истина</i> , при отжатии – <i>Ложь</i> .
Кнопка <i>Меньше</i>	При нажатии выходу <i>Меньше</i> присваивается <i>Истина</i> , при отжатии - <i>Ложь</i> .
Надпись <i>Блокировка</i>	Появляется если соответствующий вход равен <i>Истина</i> . При этом кнопки <i>Открыть</i> , <i>Закрыть</i> , <i>Автоматический</i> , <i>Дистанционный</i> становятся недоступными.
Число <i>Положение</i>	Отображается величина входа <i>Положение</i> .
Столбиковая диаграмма <i>Положение</i>	Закрашивается зеленым в зависимости от значения входа <i>Положение</i> на соответствующее число процентов.
Кнопка <i>Признаки достоверности</i>	Открывает окно <i>Признаки качества входов</i> .

Функциональный блок *Регулирующий клапан с импульсным управлением* способен в режиме исполнения выводить информацию о достоверности данных на его входах. Данные считаются недостоверными если:

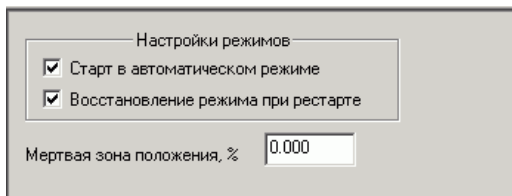
- Хотя бы один из входов ФБ имеет *Плохой* признак качества;
- Данные входов *Открыт*, *Закрыт* и *Положение* не соответствуют, т.е. если выполняется условие: $((\text{Вход Открыт} = \text{Истина}) \text{ И } (\text{Положение} < 100\% - \text{м.з.})) \text{ ИЛИ } ((\text{Вход Закрыт} = \text{Истина}) \text{ И } (\text{Положение} > \text{м.з.}))$, где м.з. - величина мертвой зоны, устанавливаемая на странице свойств ФБ *Настройка*.

Окно признаков недостоверности данных входов может быть открыто в режиме исполнения при нажатии кнопки *Признаки достоверности* в окне оператора.



Кнопка *Квитировать* - при нажатии на эту кнопку изображение в мнемосхеме перестает быть бледным вплоть до появления новых признаков недостоверности.

Вид окна страницы свойств представлен на рисунке ниже.

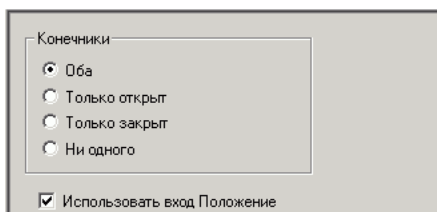


Группа *Настройка режимов* - В этом поле устанавливается, в какой режим должен перейти ФБ при старте режима исполнения:

- *Восстановление режима при рестарте* - если флаг установлен, то при последующем запуске системы будет восстановлен тот режим, в котором ФБ работал на момент останова режима исполнения в предыдущей сессии.
- *Старт в автоматическом режиме* - если флаг установлен, то включается *Автоматический режим управления*, иначе – *Дистанционный*.

Поле *Мертвая зона положения* позволяет установить зону нечувствительности положения исполнительного механизма используемую для определения состояния ФБ (Открыт/Закрыт).

Страница свойств *Состав Регулирующего клапана с импульсным управлением* представлена на рисунке ниже.



Группа *Конечники* - с помощью переключателей этой группы задается наличие входов функционального блока, принимающих сигналы от конечных выключателей (*Входы Открыт* и *Закрыт*):

- *Оба* - используются оба входа *Открыт* и *Закрыт*;
- *Только открыт* - используется только вход *Открыт*;
- *Только закрыт* - используется только вход *Закрыт*;
- *Ни одного* - входы *Открыт* и *Закрыт* не используются.

Флаг *Использовать вход Положение* - если флаг не установлен, вход *Положение* отсутствует.

Изображение ВФБ имеет две закрашиваемых области: основную и сигнальную. Основная область служит для информирования оператора о состоянии исполнительного механизма (Открыт/Закрыт). Сигнальная область предназначена для отображения аварии и переходного состояния. В режиме разработки сигнальная область отображается красным цветом.



Основной цвет формируется в зависимости от значений входов *Открыт* и *Закрыт* как указано в таблице ниже.

	Имя	Значение			
Входы	Открыт	Истина	Истина	Ложь	Ложь
	Закрыт	Истина	Ложь	Истина	Ложь
Основной цвет (цвет корпуса)		Настраивается	Настраивается	Серый	Серый

Кроме того, на странице свойств *Вид* функционального блока, можно настроить такие свойства, как *Тип механизма*, *Ориентация в мнемосхеме*, *Основной цвет*.

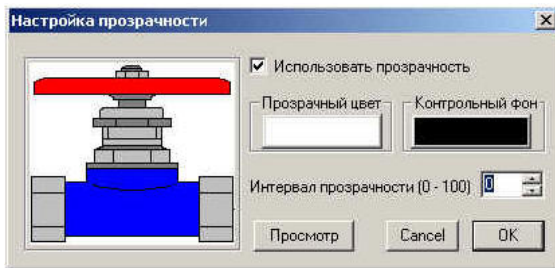
Сигнальный цвет определяется в зависимости от состояния входов *Открыт*, *Закрыт* и *Авария* в соответствии с нижеприведенной таблицей:

		Истина Выход Авария	Ложь Выход Авария
Входы	Если вход <i>Открыт</i> не равен входу <i>Закрыт</i>	Красный (если не квитированная Авария -мигающий красный)	Серый
	Если вход <i>Открыт</i> равен входу <i>Закрыт</i>	Красный (если не квитированная Авария -мигающий красный)	Желтый

Настройка изображения в мнемосхеме осуществляется аналогично настройке ФБ *Задвижка*.

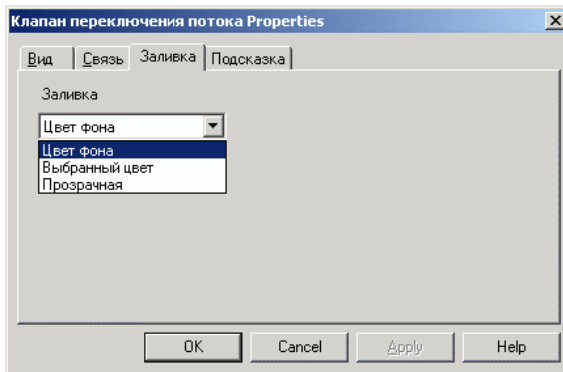
Общие свойства и настройки

В диалоге *Настройки вида* можно выбрать цвет, который не будет выводиться при отображении картинки, а также интервал прозрачности.






Если в *Окне предпросмотра* щелкнуть мышкой на изображении, то цвет, над которым находился курсор в момент щелчка, будет назначен как прозрачный. Снятие флага *Использовать прозрачность* отменяет использование прозрачности. Кнопка *Прозрачный цвет* открывает стандартный диалог выбора цвета, который будет использоваться в качестве прозрачного, т.е. не отображаться. Кнопка *Контрольный фон* открывает стандартный диалог выбора цвета, который в режиме предпросмотра будет заменять прозрачный. Если необходимо, чтобы прозрачным был не только выбранный цвет, но и близкие к нему, следует установить величину *Интервал прозрачности* в соответствующее значение (0 - строго выбранный цвет, 100 - в интервал попадут все цвета). Сначала нужно выбрать некоторый прозрачный цвет, затем включить режим предпросмотра и увеличивать интервал прозрачности до тех пор, пока не исчезнут все нежелательные оттенки. Кнопка *Просмотр* включает режим предпросмотра, в котором *Прозрачный цвет* заменяется на цвет контрольного фона.

Окно заливки фона имеет следующий вид:



Существует 3 вида заливки фона:

-  Заливка цветом фона мнемосхемы;
-  Заливка выбранным вами цветом;
-  Прозрачная (нет заливки).

Датчики

Индикатор графический

Индикатор графический является простым визуальным функциональным блоком и предназначен для индикации состояния дискретных переменных. Индикатор графический не представлен ни в палитре функциональных блоков MasterSCADA, ни в палитре элементов графического редактора. Единственный способ, с помощью которого можно создать элемент управления *Индикатор графический* на мнемосхеме - перетащить дискретную переменную (*Значение*, *Расчет*, *Событие*, *ОПС переменная для чтения*, *Выход функционального блока*) в мнемосхему, для того чтобы наблюдать состояние этой переменной.

Для *Индикатора графического* существует возможность выбора графических файлов изображений для каждого из состояний связанной дискретной переменной (*Истина*, *Ложь*). Эти настройки производятся на странице свойств *Вид*. ВФБ *Индикатор графический* имеет возможность отображать недостоверность данных входа.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Индикатор мнемонический

Индикатор мнемонический является простым визуальным функциональным блоком и предназначен для индикации состояния дискретных переменных. Индикатор мнемонический не представлен ни в палитре функциональных блоков MasterSCADA, ни в палитре элементов графического редактора. Единственный способ, с помощью которого можно создать элемент управления Индикатор мнемонический на мнемосхеме - перетащить дискретную переменную (*Значение, Расчет, Событие, OPC переменная для чтения, Выход функционального блока*) в мнемосхему, для того чтобы наблюдать состояние этой переменной.

Для *Индикатора мнемонического* существует возможность задать цвет ее отображения в зависимости от состояния связанной дискретной переменной. Эти и другие настройки внешнего вида производятся на странице свойств *Вид*. ВФБ *Индикатор мнемонический* имеет возможность отображать недостоверность данных входа

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

График

Визуальный функциональный блок *График* относится к категории *Датчики* и имеет настраиваемое количество входов, соответствующих параметрам графика.

Вид функционального блока в дереве объекта:

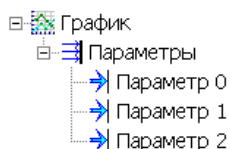


График предназначен для просмотра значений переменных в мнемосхеме. Существует три основных способа создания элемента управления график в мнемосхеме:

- Вставить ВФБ *График* из палитры функциональных блоков в объект, привязать интересующие переменные к входам графика, и перетащить его в мнемосхему;
- Для просмотра значений аналоговых переменных в виде графика (*Значение, Расчет, OPC переменная для чтения, Выход ФБ*), можно просто перетащить их в мнемосхему правой кнопкой мыши, и в открывающемся контекстном меню выбрать *График*;
- Создать элемент управления *График* непосредственно в мнемосхеме. Затем, завести необходимое количество входов мнемосхемы, связанных с переменными MasterSCADA, и динамизировать свойства графика *Значение* от этих входов.

График реализует функции тренда реального времени (зависимость параметра от времени). Может иметь от 1 до 8 параметров.



Диапазон шкалы, единица измерения и формат значения каждого параметра наследуются от настроек переменной, привязанной к параметру графика в дереве объекта. Число параметров, отображаемых на графике задается в проекте на странице свойств *Свойства*. По умолчанию это всегда один параметр. ВФБ график имеет возможность отображать недостоверность данных входов

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Индикатор состояния

Индикатор состояния имеет группу логических входов, число которых настраивается. Каждому входу сопоставлен некоторый цвет, которым заполняется изображение блока в мнемосхеме. Текущим входом считается тот, у которого переход из *Ложь* в *Истина* (передний фронт) произошел последним (если перешли больше одного одновременно, текущим будет тот, который с наибольшим номером). Если после старта ни одного входа не перешли, цвет будет по умолчанию. ВФБ *Индикатор состояния* имеет возможность отображать недостоверность данных входов

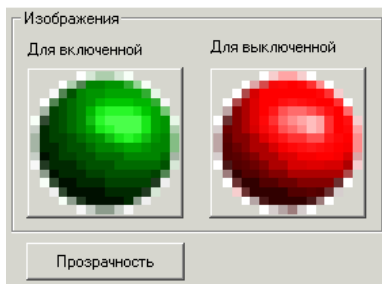
Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Свойства Индикаторов

Для изображения *Индикатора графического* необходимо выбрать два файла с растровыми изображениями (*BMP, GIF, JPG*). Одно изображение - для включенного состояния, другое - для выключенного, возможно оставить изображения, имеющиеся по умолчанию.

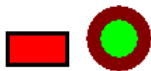


У изображения *Индикатора графического* в мнемосхеме есть свойства, которые можно настроить. Для этого выберите пункт *Свойства* в контекстном меню, всплывающем при нажатии правой клавиши мыши на изображении.

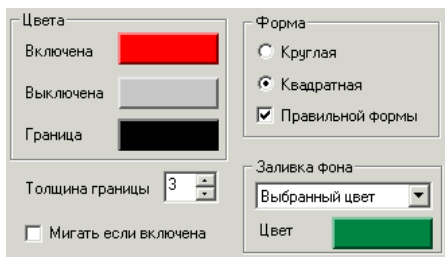


При нажатии на изображения откроется диалог выбора пользовательского изображения.

Индикатор мнемонический изображается простой геометрической фигурой - кругом или квадратом с настраиваемыми цветами.



У изображения *Индикатора мнемонического* есть свойства, которые можно настроить. Для этого выберите пункт *Свойства* в контекстном меню, всплывающем при нажатии правой клавиши мыши на изображении.



Толщина границы задается в пикселях. Группа *Цвета* содержит цветные кнопки для выбора цветов. Форма может быть следующих типов:

- *Круглая* - круглая (или эллиптическая) форма;
- *Квадратная* - квадратная (или прямоугольная) форма;
- *Правильной формы* - если установлен этот флаг, то будет точный круг или квадрат, если не установлен - эллипс или прямоугольник, растягивающийся по размерам рамки.

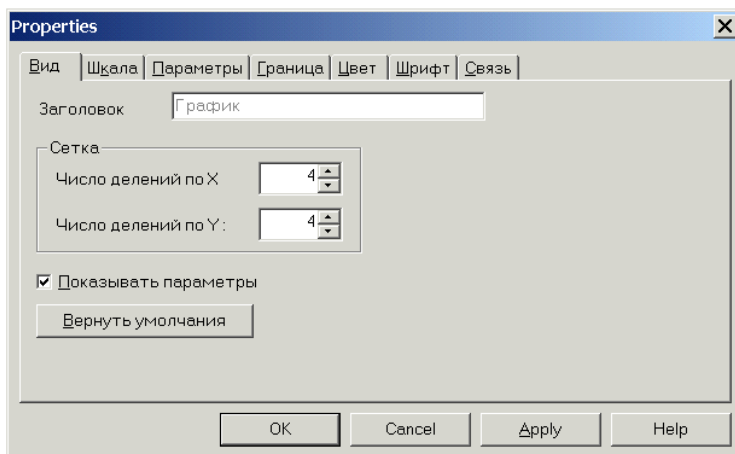
Если установлен флаг *Мигать если включена*, то во включенном состоянии лампочка будет менять цвет. Если не установлен - во включенном состоянии лампочка будет одного цвета (заданного для состояния *Включена*). На вид в выключенном состоянии этот флаг не влияет.

Заливка фона бывает трёх видов:

- *Общий цвет фона* - свободное пространство заливается цветом, выбранным для фона мнемосхемы;
- *Выбранный цвет* - цвет заливки свободного пространства выбирается кнопкой (кнопка доступна только для этого способа);
- *Прозрачная* - свободное пространство не заливается.

Свойства Графика

На закладке *Вид* настраивается внешний вид ВФБ *График*.

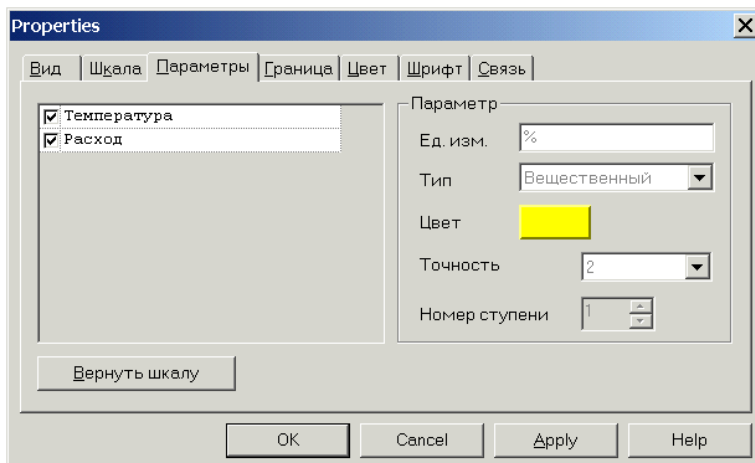


Поле ввода *Заголовок* - текст заголовка наследуется от имени данного экземпляра ВФБ *График* в дереве объекта (название *График* в дереве объекта может быть изменено), новое имя может быть задано в поле ввода, при отсутствии текста заголовка не имеет. Поле ввода *Число делений по X* задает число делений графика по оси X.

Поле ввода *Число делений по Y* задает число делений графика по оси Y. Флаг *Показывать параметры* - задает необходимость отображения таблицы параметров под графиком (таблица содержит для каждого параметра имя, текущее значение и единицу измерения). Кнопка *Вернуть умолчания* восстанавливает свойства, унаследованные из дерева объекта

- ❖ **Важно!** При отсутствии связи у выхода ВФБ или работы графического редактора вне среды пакета программ MasterSCADA наследование настроек свойств переменной не работает. В этом случае все свойства графика настраиваются непосредственно на страницах свойств. После установления в дереве связи выхода ВФБ для импорта всех необходимых настроек следует выполнить команду меню *Сервис - Обновить связи*.

На закладке *Параметры* задаются настройки отображения параметров ВФБ *График*.



В списке отображаемых параметров флажками задаются те параметры, которые должны отображаться на графике, название параметра может быть изменено (Внимание! При привязке переменных к параметрам графика в дереве объекта необходимо начинать с первого параметра). В списке отображаемых параметров флагами задаются те параметры, которые должны отображаться на графике, название параметра может быть изменено.

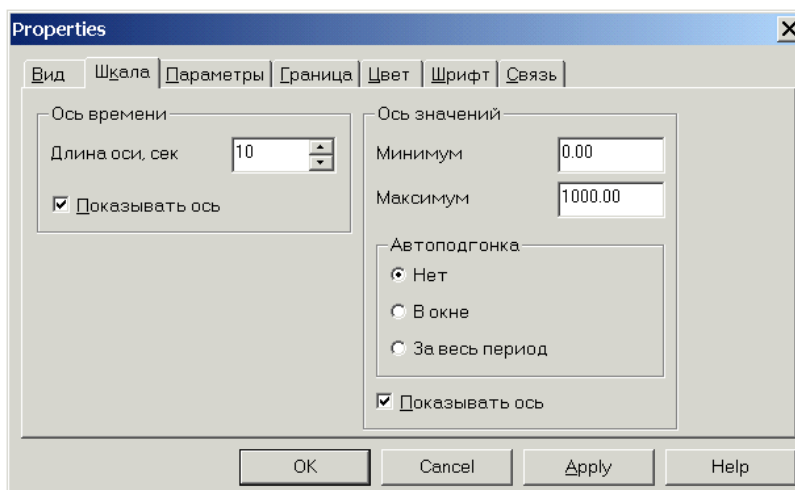
- ❖ **Важно!** При привязке переменных к параметрам графика в дереве объекта необходимо начинать с первого параметра.

Функции элементов управления:

- Поле ввода *Ед. изм.* - единица измерения наследуется из свойств переменной дерева объекта и может быть изменена, при отсутствии текста единица измерения не пишется;
- Поле ввода *Тип* - тип значения наследуется из свойств переменной дерева объекта и может быть изменен, при этом будет производиться необходимое преобразование значения;
- Кнопка *Цвет* - задает цвет, которым будет отображаться график выбранного в списке параметра;
- Выпадающий список *Точность* - точность представления числа (число знаков после запятой) наследуется из свойств переменной дерева объекта и может быть изменена;
- Поле ввода *Номер ступени* используется для разнесения по вертикали графиков нескольких дискретных параметров (чтобы избежать наложение их друг на друга), задает ступень параметра;
- Кнопка *Вернуть шкалу* восстанавливает свойства, унаследованные из дерева объекта.



На закладке *Шкала* настраиваются оси графика ВФБ.



Функции элементов управления:

- Поле ввода *Длина оси, сек* задает диапазон оси времени графика в секундах;
- Флаг *Показывать ось* задает необходимость отображения оси времени на поле графика;
- Поле ввода *Минимум* задает значение нижней границы диапазона шкалы значений;
- Поле ввода *Максимум* задает значение верхней границы диапазона шкалы значений;

- Поле выбора *Автоподгонка* задает вид отображения диапазона шкалы по оси значений:
 - *Нет* означает, что диапазон шкалы соответствует заданным границам;
 - *В окне* означает, что диапазон шкалы автоматически выбирается от минимума до максимума значения параметра на видимом в настоящий момент в окне отрезке графика;
 - *За весь период* означает, что диапазон автоматически выбирается от минимума до максимума значения параметра за все время отображения графика на мнемосхеме (с момента последнего открытия мнемосхемы);
- Флаг *Показывать ось* задает необходимость отображения оси значений на поле графика.

Аппараты

Емкость

ВФБ *Емкость* относится к категории *Аппараты* и имеет три входа: *Уровень*, *Температура* и *Авария*.



Основное предназначение этого элемента - визуализация состояния технологического аппарата типа емкость, оборудованного аналоговыми датчиками уровня и температуры жидкости. При поступлении сигнала *Авария*, изображение *Емкости* начинает мигать. Сигнал *Авария* может быть сформирован внешними средствами, например, с помощью расчетной переменной. При поступлении значений на вход *Уровень*, на мнемосхеме отображается уровень заполнения *Емкости* соответственно входному значению. При поступлении значений на вход *Температура* на мнемосхеме меняется цвет заполнения *Емкости* соответственно входному значению от синего к красному (по умолчанию).

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Емкость с мешалкой

ВФБ *Емкость с мешалкой* относится к категории *Аппараты* и имеет четыре входа: *Уровень*, *Температура*, *Авария* и *Мешалка*.

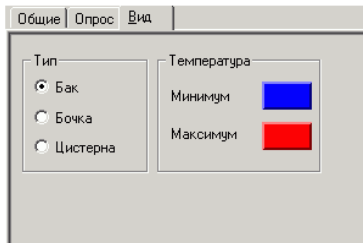





Основное предназначение этого элемента - визуализация состояния технологического аппарата типа емкость, оборудованного мешалкой и аналоговыми датчиками уровня и температуры жидкости. При поступлении сигнала *Авария*, изображение *Емкости* начинает мигать. Сигнал *Авария* может быть сформирован внешними средствами, например, с помощью расчетной переменной. При поступлении значений на вход *Уровень*, на мнемосхеме отображается уровень заполнения *Емкости* соответственно входному значению. При поступлении значений на вход *Температура* на мнемосхеме меняется цвет заполнения *Емкости* соответственно входному значению от синего к красному (по умолчанию). При поступлении сигнала *Мешалка* - изображение мешалки имитирует ее вращение.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Страницы свойств ФБ

На закладке *Вид* настраивается внешний вид ВФБ *Емкость*.

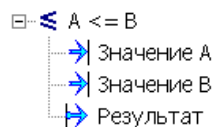


Имеется возможность выбора типа *Емкости* в виде *Бака* , *Бочки*  или *Цистерны* . В группе температура возможно установить цвета жидкости в *Емкости* при минимальной и максимальной температуре. Промежуточные значения будут отображаться промежуточным цветом. При нажатии на цветную кнопку всплывает стандартный диалог *Windows* для выбора цвета.

Вычисления

Меньше или равно

Если *Значение A* \leq *Значение B*, то *Результат* - *Истина*, иначе – *Ложь*. Вид функционального блока в дереве объекта:



Входы и выходы ФБ:

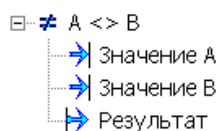
	Обозначение	Тип
Входы	Значение A	Вещественный дв. точности
	Значение B	Вещественный дв. точности
Выход	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Неравно

Если *Значение A* \neq *Значение B*, то *Результат* - *Истина*, иначе – *Ложь*.

Вид функционального блока в дереве объекта:



Входы и выходы ФБ:

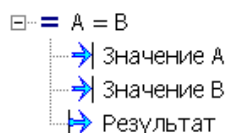
	Обозначение	Тип
Входы	Значение A	Вещественный дв. точности
	Значение B	Вещественный дв. точности
Выход	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Равно

Если *Значение A* = *Значение B*, то *Результат* - *Истина*, иначе – *Ложь*.

Вид функционального блока в дереве объекта:



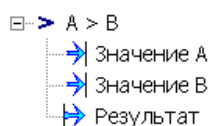
Входы и выходы ФБ:

	Обозначение	Тип
Входы	Значение А	Вещественный дв. точности
	Значение В	Вещественный дв. точности
Выход	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Больше

Если *Значение А > Значение В*, то *Результат - Истина*, иначе – *Ложь*. Вид функционального блока в дереве объекта:



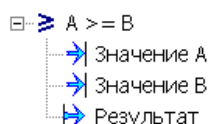
Входы и выходы ФБ:

	Обозначение	Тип
Входы	Значение А	Вещественный дв. точности
	Значение В	Вещественный дв. точности
Выход	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Больше или равно

Если *Значение А >= Значение В*, то *Результат - Истина*, иначе – *Ложь*. Вид функционального блока в дереве объекта:



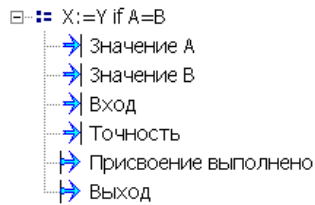
Входы и выходы ФБ:

	Обозначение	Тип
Входы	Значение А	Вещественный дв. точности
	Значение В	Вещественный дв. точности
Выход	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

X:=Y if A=B

Условная операция. На *Выходе* будет значение, подаваемое на *Вход*, если *Значение А* и *Значение В* различаются строго меньше чем на величину *Точность* то есть $|A-B| < \text{Точность}$. В противном случае, на *Выходе* будет ноль. Кроме того, в случае удачного выполнения присваивания на выходе *Присвоение* выполнено будет *Истина*, иначе – *Ложь*. Метка времени и признаки качества входа передаются на выход. Обязательно к заполнению – *Вход Точность*. Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется. Вид функционального блока в дереве объекта:



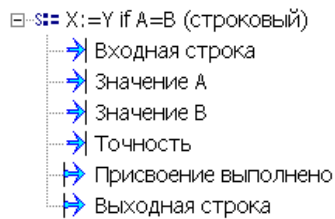
Входы и выходы:

	Обозначение	Тип
Входы	Значение А	Вещественное
	Значение В	Вещественное
	Вход	Вещественное
	Точность	Вещественное
Выходы	Присвоение выполнено	Логическое
	Выход	Вещественное

X:=Y if A=B (строковое)

Условная операция. На *Выходе* будет значение, подаваемое на *Вход*, если *Значение А* и *Значение В* различаются строго меньше чем на величину *Точность* то есть $|A-B| < \text{Точность}$. В противном случае, на *Выходе* будет ноль. Кроме того, в случае удачного выполнения присваивания на выходе *Присвоение выполнено* будет *Истина*, иначе – *Ложь*. Метка времени и признаки качества входа передаются на выход.

Вид функционального блока в дереве объекта:



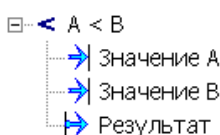
Входы и выходы:

	Обозначение	Тип
Входы	Значение А	Вещественное
	Значение В	Вещественное
	Входная строка	Строковое
	Точность	Вещественное
Выходы	Присвоение выполнено	Логическое
	Выход	Строковое

Обязательно к заполнению – *Вход Точность*. Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Меньше

Если *Значение А < Значение В*, то Результат - *Истина*, иначе – *Ложь*. Вид функционального блока в дереве объекта:



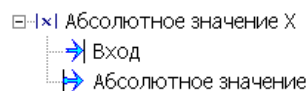
Входы и выходы ФБ:

	Обозначение	Тип
Входы	Значение А	Вещественный дв. точности
	Значение В	Вещественный дв. точности
Выход	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Абсолютное значение X

Вычисляет абсолютное значение (модуль) *Входа*. Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется. Вид функционального блока в дереве объекта:

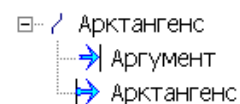


Входы и выходы:

	Обозначение	Тип
Вход	Вход	Вещественный двойной точности
Выход	Абсолютное значение	Вещественный двойной точности

Арктангенс X

Вычисляет арктангенс значения Аргумента. Вид функционального блока в дереве объекта:



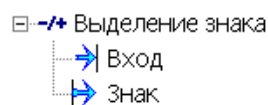
Входы и выходы:

	Обозначение	Тип
Вход	Аргумент	Вещественный двойной точности
Выход	Арктангенс	Вещественный двойной точности

Результат на выходе - в радианах. Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Выделение знака

Если на *Входе* – отрицательное значение, на выходе *Знак* = -1. Если на *Входе* – положительное значение, на выходе *Знак* = 1. Если на *Входе* – нулевое значение, на выходе *Знак* = 0. Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Вход	Вход	Вещественный двойной точности
Выход	Знак	Вещественный двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Выделение подстроки

Функциональный блок *Выделение подстроки* предназначен для выделения нескольких идущих подряд символов из одного массива символов.

Входы:

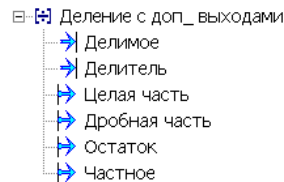
- *Строка* - задается исходная строка;
- *Индекс начала подстроки* - первый символ подстроки (порядковый номер символа), которую нужно выделить;
- *Длина подстроки* - определяется количество символов в подстроке.

Выходы:

- *Подстрока* – результат.

Деление с дополнительными выходами

Деление с формированием дополнительных выходов: целой и дробной частей, остатка от деления нацело. Вид функционального блока в дереве объекта:



Входы и выходы:

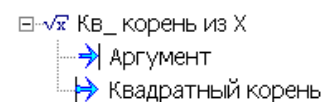
	Обозначение	Тип
Вход	Делимое	Вещественный
	Делитель	Вещественный
	Целая часть	Целый
Выход	Дробная часть	Вещественный
	Остаток	Целый
	Частное	Вещественный

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Квадратный корень из X

Вычисляет квадратный корень положительного *Аргумента*. При отрицательных значениях *Аргумента* на выходе остается результат предыдущего вычисления.

Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Вход	Аргумент	Вещественный двойной точности
Выход	Квадратный корень	Вещественный двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Конкатенация строк

Служит для сложения нескольких строк в одну. Количество строк определяется на странице свойств *Настройка* данного функционального блока.

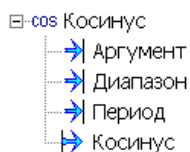
Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Косинус

Масштабирует и вычисляет косинус *Аргумента* по формуле:

$$\text{Косинус} = \frac{\text{Диапазон}}{2} \cos \frac{2 * \pi * \text{Аргумент}}{\text{Период}}$$

Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Входы	Аргумент	Вещественное двойной точности
	Диапазон	Вещественное двойной точности
	Период	Вещественное двойной точности
Выход	Косинус	Вещественное двойной точности

Обязательно к заполнению – *Вход Период*. Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Логическое И

Логический оператор. Вид функционального блока в дереве объекта:

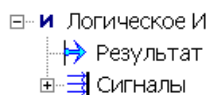


Таблица истинности:

X	Y	Результат
Истина	Истина	Истина
Истина	Ложь	Ложь
Ложь	Истина	Ложь
Ложь	Ложь	Ложь

Настройки:

Свойство	Тип свойства	Значение
Число операндов	Целое	От 2 до 255

Входы и выходы:

	Обозначение	Тип
Входы	Сигнал 1	Логический

	Сигнал N	Логический
Выходы	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Логическое ИЛИ

Логический оператор. Вид функционального блока в дереве объекта:

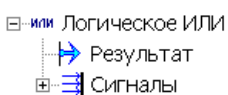


Таблица истинности:

X	Y	Результат
---	---	-----------

Истина	Истина	Истина
Истина	Ложь	Истина
Ложь	Истина	Истина
Ложь	Ложь	Ложь

Настройки:

Свойство	Тип свойства	Значение
Число операндов	Целое	От 2 до 255

Входы и выходы:

	Обозначение	Тип
Входы	Сигнал 1	Логический

	Сигнал N	Логический
Выходы	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Логическое Исключающее ИЛИ

Логический оператор. Вид функционального блока в дереве объекта:

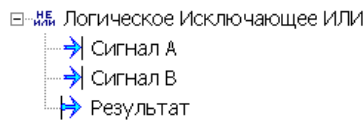


Таблица истинности:

X	Y	Результат
Истина	Истина	Истина
Истина	Ложь	Ложь
Ложь	Истина	Ложь
Ложь	Ложь	Истина

Входы и выходы:

	Обозначение	Тип
Входы	Сигнал А	Логический
	Сигнал В	Логический
Выходы	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Логическое НЕ

Логический оператор. Вид функционального блока в дереве объекта:

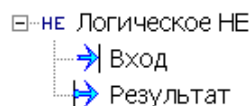


Таблица истинности:

X	Результат
Истина	Ложь
Ложь	Истина

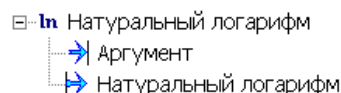
Входы и выходы:

	Обозначение	Тип
Входы	Вход	Логический
Выходы	Результат	Логический

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Натуральный логарифм

Логарифмирование по основанию e. Вид функционального блока в дереве объекта:



Входы и выходы:

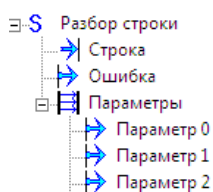
	Обозначение	Тип
Входы	Аргумент	Вещественный двойной точности
Выходы	Натуральный логарифм	Вещественный двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Разбор строки

Функциональный блок *Разбор строки* позволяет выделять отдельные числовые параметры, если несколько этих параметров приходит упакованными в строку. Предназначение ФБ *Разбор строки* аналогично ФБ *Выделение подстроки*. Существенным отличием является то, что тип параметров рассматриваемого функционального блока можно менять и, соответственно, использовать в дальнейшем в дереве объектов.

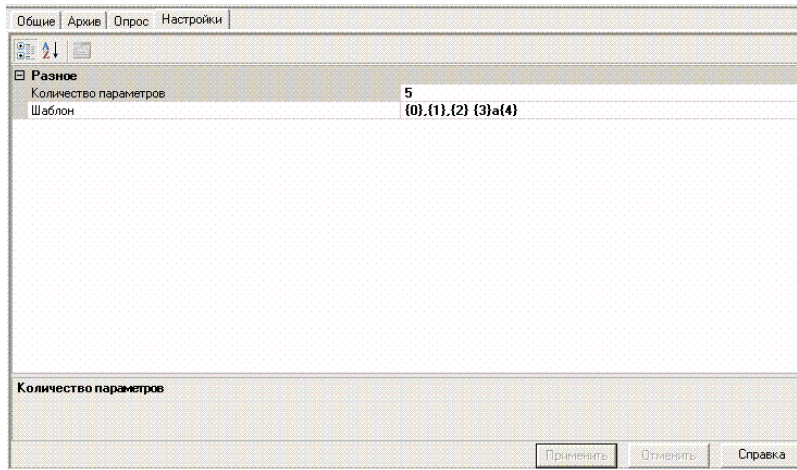
Вид функционального блока в дереве Объекта:



Входы и выходы:

	Имя	Тип	Примечание
Вход	Строка	Строковый	
	Ошибка	Дискретный	Если не указан шаблон или строка не соответствует шаблону принимает значение <i>Истина</i> .
Выходы	Параметры	Группа переменных	Количество переменных <i>Параметр</i> определяется на закладке <i>Настройки</i> панели свойств ФБ в строке <i>Количество параметров</i> . Каждая переменная имеет свой идентификационный номер, например, <i>Параметр 0</i> , <i>Параметр 1</i> и т. д.

Страница свойств *Настройки* выглядит следующим образом.

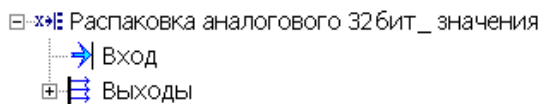


В поле *Количество параметров* определяется, сколько выходов будет иметь функциональный блок. Количество параметров, используемых в поле *Шаблон* должно совпадать с числом параметров, назначаемых в этой строке. В поле *Шаблон* задаётся шаблон разбора строки, формируемой на входе, вида {0} {1} {2}. В фигурных скобках указывается идентификационный номер параметра, которому будет присвоено значение, выделенное из строки в том месте, в котором он стоит. Примечание: "." необходимо использовать в качестве разделителя для вещественных чисел. Например, если настроить ФБ Разбор строки так, как показано на иллюстрации страницы свойств *Настройки*, а на закладке *Опрос* страниц свойств входа *Строка* в поле *Константа* ввести следующее выражение 33.3,33.3,33.3 33.3a33.3, то при запуске проекта на исполнение получим следующий результат:

Разбор строки	Строка	33.3,33.3,33.3 33.3a33.3
Параметры	Параметр 0	33.300
	Параметр 1	33.300
	Параметр 2	33.300
	Параметр 3	33.300
	Параметр 4	33.300

Распаковка аналогового 32-х битового значения

На 32 *Выхода* подаются значения каждого бита числа, подаваемого на *Вход* в соответствующей позиции двоичной записи. Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Входы	Вход	Целый
	Выход 0	Логический
Выходы
	Выход 31	Логический

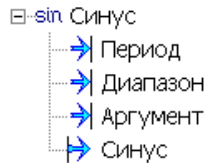
Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Синус

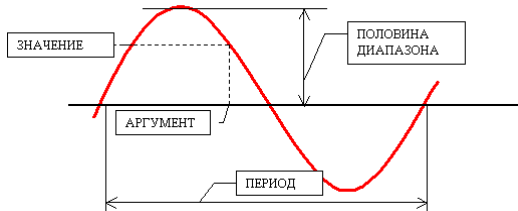
Масштабирует и вычисляет синус *Аргумента* по формуле:

$$\text{Синус} = \frac{\text{Диапазон}}{2} \sin\left(\frac{2 \times \pi \times \text{Аргумент}}{\text{Период}}\right)$$

Вид функционального блока в дереве объекта:



Из формулы следует, в частности, что амплитуда синуса будет равна половине значения *Диапазон*. Если вы хотите вычислять аргумент, в градусах, необходимо установить *Период* = 360, для аргумента, заданного в радианах необходимо установить *Период* = 2π . Графический смысл параметров:



Если необходимо использовать функцию в качестве генератора сигнала синусоидальной формы, на вход *Аргумент* необходимо подать время. Для этого можно воспользоваться, например функциональным блоком *Счетчик пробега*.

Входы и выходы:

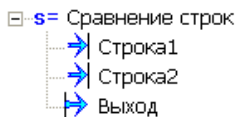
	Обозначение	Тип
Входы	Период	Вещественное двойной точности
	Диапазон	Вещественное двойной точности
	Аргумент	Вещественное двойной точности
Выход	Синус	Вещественное двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Сравнение строк

Осуществляет сравнение значений поступающих на *Входы Строка 1* и *Строка 2*. В случае равенства входных значений на *Выходе* устанавливается логическая единица.

Вид функционального блока в дереве объекта:



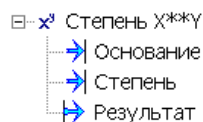
Входы и выходы ФБ:

	Имя	Тип	примечание
Входы	Строка 1	Строковый	Первое сравниваемое значение
	Строка 2	Строковый	Второе сравниваемое значение
Выход	Выход	Логический	Принимает значение логической единицы в случае равенства <i>Строки 1</i> и <i>Строки 2</i> .

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Степень X**Y

Возведение положительного числа *Основание* в степень *Степень*. При отрицательных значениях *Основания* результатом возведения будет ноль. Вид функционального блока в дереве объекта:



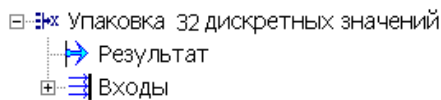
Входы и выходы:

	Обозначение	Тип
Входы	Основание	Вещественный двойной точности
	Степень	Вещественный двойной точности
Выход	Результат	Вещественный двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Упаковка 32 дискретных значений

Из 32-х логических значений на входе создает значение целого типа, используя значения *Истина* и *Ложь* в качестве 1 и 0 в соответствующей позиции 32-значного двоичного числа. Вид функционального блока в дереве объекта:



Входы и выходы:

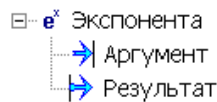
	Обозначение	Тип
Входы	Вход 1	Логический

	Вход 3	Логический
Входы	Выход	Целый

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Экспонента X

$\exp(x)$ - возводит число e в степень *Аргумент*. Вид функционального блока в дереве объекта:



Входы и выходы:

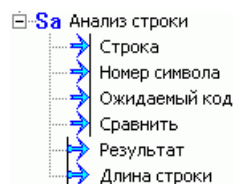
	Обозначение	Тип
Входы	Аргумент	Вещественный двойной точности
Выходы	Результат	Вещественный двойной точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Анализ строки

ФБ *Анализ строки* служит для сравнения *ASCII* кода конкретного символа строки с заданным кодом и для вычисления длины всей строки.

Вид функционального блока в дереве объекта:



Входы/Выходы:

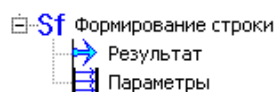
	Обозначение	Тип	Описание
Входы	Строка	Строковый	Анализируемая строка.
	Номер символа	Целый	Номер анализируемого символа в строке. Нумерация символов в строке начинается с 0.
	Ожидаемый код	Целый	Десятичный ASCII код.
	Сравнить	Логический	При значении <i>Вкл</i> ФБ производит анализ: сравнивает код символа строки, номер которого указан на входе Номер символа, с кодом, заданным на входе <i>Ожидаемый код</i> , и результат сравнения выдает на выход <i>Результат</i> ; считает длину строки и выдает это значение на выход <i>Длина строки</i> . При значении <i>Выкл</i> ФБ не производит никаких расчетов.
Выходы	Результат	Логический	Результат сравнения, выполненного при значении <i>Вкл</i> на входе <i>Сравнить</i> , кода символа строки, номер которого указан на входе Номер символа, с кодом, заданным на входе <i>Ожидаемый код</i> .
	Длина строки	Целый	Длина строки, рассчитанная при значении <i>Вкл</i> на входе <i>Сравнить</i> .

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Формирование строки

Функциональный блок *Формирование строки* предназначен для динамического формирования текстовой строки в режиме исполнения.

Вид функционального блока в дереве *Объекта*:

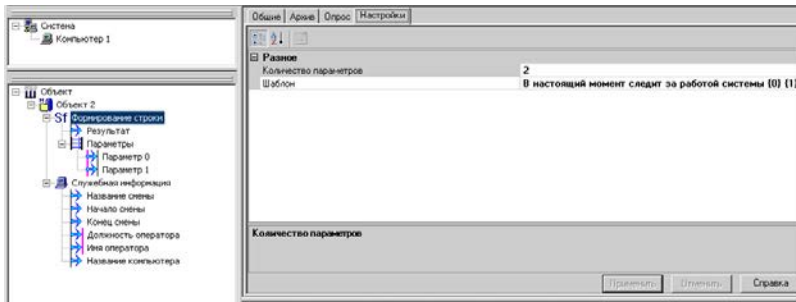


Входы и выходы:

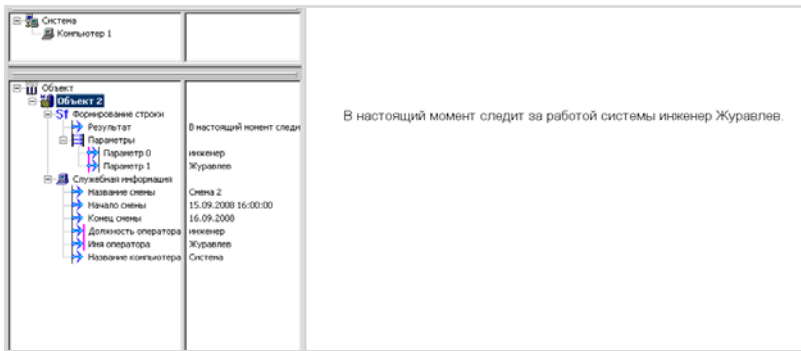
	Имя	Тип	Примечание
Входы	Параметры	Группа переменных	Количество переменных <i>Параметр</i> определяется на закладке <i>Настройки</i> панели свойств ФБ <i>Формирование строки</i> . Каждая переменная имеет свой идентификационный номер, например, <i>Параметр 0</i> , <i>Параметр 1</i> и т. д.
	Выходы	Результат	Строковый

Страница свойств *Настройки*:

В поле *Шаблон* вводится строка, которая будет отображаться на выходе функционального блока *Результат* в режиме исполнения. Значения входных переменных *Параметр* могут так же отображаться в формируемой строке в том месте, где в шаблон будет вставлен идентификационный номер переменной *Параметр* в фигурных скобках *{}*. Например, если настроить ФБ *Формирование строки* так, как показано на рисунке, и добавить в дерево объекта ФБ *Службная информация*



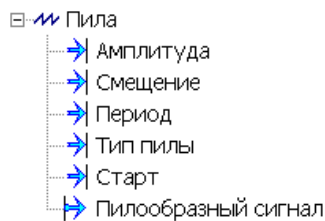
На мнемосхеме будет отображаться требуемый текст:



Генераторы значений

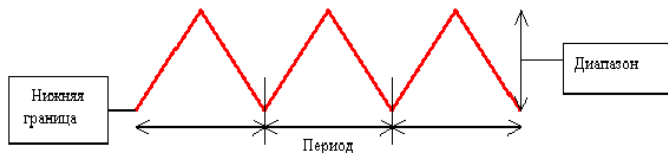
Пила

Если *Старт=Ложь*, то выход не изменяет старого значения. Вид функционального блока в дереве объекта:



При переходе *Старт* с *Ложь* на *Истина* для последующих просчетов *Пилообразный сигнал* рассчитывается как периодическая ломаная линия (с периодом *Период*) согласно значению *Тип пилы*.

Тип пилы= 0 - обычная восходяще-нисходящая пила.



Тип пилы= 1 - восходящая пила.



Тип пилы= 2 - нисходящая пила.



Изменение *Периода* или *Типа пилы* во время работы равносильно переходу *Старт* с *Ложь* на *Истина*.

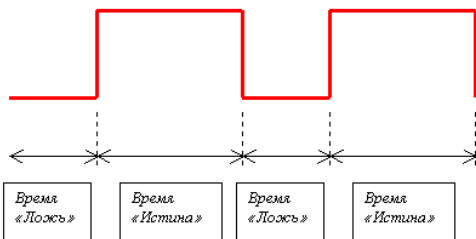
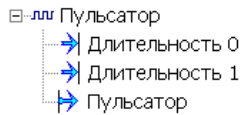
Входы: *Диапазон (Вещественный)*, *Нижняя граница (Вещественный)*, *Период (Вещественный)* *Тип пилы (Целый)*, *Старт(Логический)*.

Выходы: *Пилообразный сигнал (Вещественный)*

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Пульсатор

Функциональный блок генерирует логические значения, изменяющиеся с заданным периодом. Начальное значение – *Ложь*. Вид функционального блока в дереве объекта:



Входы и выходы:

	Обозначение	Тип
Входы	Длительность 0	Вещественный
	Длительность 1	Вещественный
Выходы	Результат	Логический

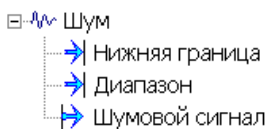
Если блок опрашивается раньше, чем должно произойти ближайшее переключение (время Переключения рассчитывается по формуле, приведенной ниже), то значение выхода не меняется, иначе блок действует по следующему алгоритму:

- Значение выхода меняется на противоположное;
- Рассчитывается новое время переключения на основе значений входов, полученных при предыдущем опросе, по следующей формуле: $\text{время Переключения} = \text{Текущее время} + \text{Период}$, где *Период* - это *Длительность0*, либо *Длительность1*, в зависимости от переключения, которое необходимо произвести.

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Шум

Создает произвольное значение (случайное число) на выходе в интервале от *Нижняя граница* до *Нижняя граница + Диапазон*. Вид функционального блока в дереве объекта:



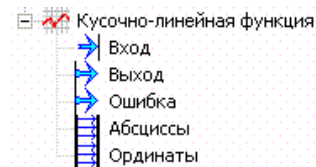
Входы и выходы:

	Обозначение	Тип
Входы	Диапазон	Вещественный
	Нижняя граница	Вещественный
Выходы	Шумовой сигнал	Вещественный

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Кусочно-линейная функция

Вид функционального блока в дереве объекта:



Управление

Двухпозиционный регулятор

Осуществляет автоматическое управление простым двухпозиционным исполнительным механизмом.

Входы и выходы ФБ:

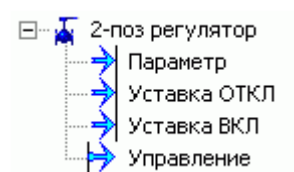
	Имя	Тип	Примечание
Вход	Параметр	вещ. дв. точн.	Сравнивается с уставкой (ами)
Вход	Уставка ОТКЛ	вещ. дв. точн.	Верхняя граница
Вход	Уставка ВКЛ	вещ. дв. точн.	Нижняя граница
Выход	Управление	логический	Сформированное значение управления

Свойства:

Имя	Тип	Примечание
Число уставок (1 или 2)	целый	Существует 2 варианта регулятора
Мертвая зона	вещ. дв. точн.	Имеет смысл только для варианта с 1-й уставкой

Алгоритм работы описан ниже.

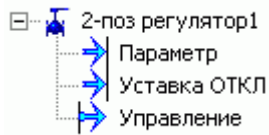
Вариант с 2-мя уставками. Если значение регулируемого параметра больше Уставки *Выкл.*, то на выходе *Управление* значение *Ложь*.



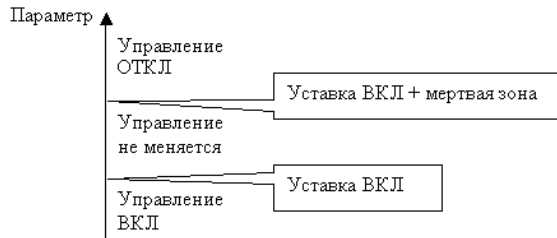
Если значение регулируемого параметра меньше этой уставки, то на выходе *Управление* значение *Истина*. Если значение параметра находится в пределах *Уставка Откл. ... Уставка Вкл.*, то значение на выходе *Управление* не меняется.



Вариант с 1-й уставкой. Если значение регулируемого параметра больше суммы *Уставки Вкл.* и мертвой зоны, то на выходе *Управление* значение *Ложь*.



Если значение регулируемого параметра меньше *Уставки Вкл.*, на выходе *Управление* значение *Истина*. Если значение параметра находится в пределах *Уставка Вкл..Уставка Вкл. + мертвая зона*, то значение на выходе *Управление* не меняется.

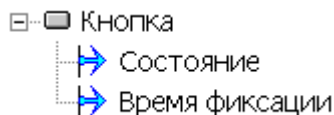


Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Кнопка

ВФБ *Кнопка* относится к категории *Управление* и имеет два выхода.

Вид функционального блока в дереве объекта



При необходимости использования ФБ с данной функциональностью необходимо использовать *Команда*. *Кнопка* - управление состоянием дискретных переменных-входов (*Значение*, *Вход ФБ*, *ОПС переменная для записи*) из мнемосхемы. Для этой цели можно сделать следующее:

- Вставить ВФБ *Кнопка* в объект из палитры функциональных блоков MasterSCADA, связать выход *Состояние* с переменной-входом, которой требуется управлять. Затем перетащить ВФБ *Кнопка* в мнемосхему;
- Вставить в объект дискретную *Команду*, связать ее с необходимым числом переменных-входов. Затем, перетащить *Команду* в мнемосхему правой кнопкой мыши, и в открывающемся контекстном меню выбрать *Кнопка*;
- Создать элемент управления *Кнопка* непосредственно в мнемосхеме. Завести выход мнемосхемы и связать его с дискретной переменной-входом, которой необходимо управлять. Далее, динамизировать свойство *Нажата Кнопки* по этому выходу.

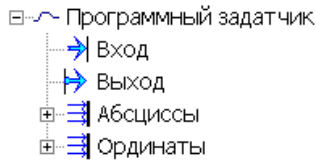
На поле кнопки можно поместить надпись и/или рисунок. Нажатому состоянию кнопки соответствует значение *Вкл.* дискретного выхода *Состояние*.



Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется. Время нажатия на кнопку должно быть больше или равно периоду опроса ФБ.

Программный задатчик

Кусочно-линейная функция. Вид функционального блока в дереве объекта:



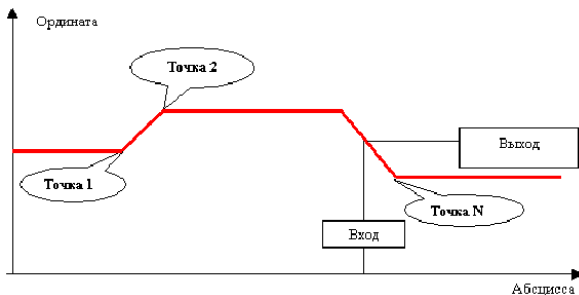
Значения:

- Если аргумент меньше, чем *Абсцисса* первой точки, то значение *Выхода* равно *Ординате* первой точки.
- Если аргумент больше, чем *Абсцисса* последней точки, то значение *Выхода* равно *Ординате* последней точки.

В остальных случаях значение *Выхода* вычисляется по аргументу в соответствии с ломаной линией, составленной из заданных точек. *Программный задатчик* используется для получения сигнала заданной формы во времени. Для этого необходимо подавать на вход равномерно возрастающую величину со скоростью изменения равной единице. Для этого можно использовать, например, *Счетчик пробега*.

Настройки:

Свойство	Тип свойства	Значение
Количество точек графика	Целое	От 1 до 128



Входы и выходы:

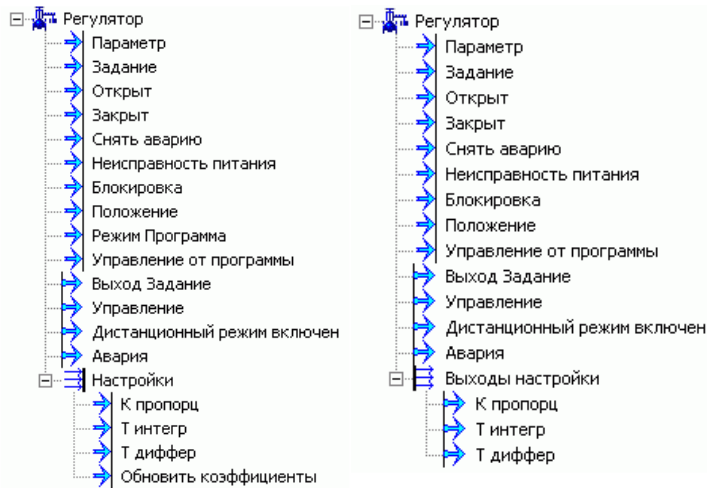
	Обозначение	Тип
Входы	Вход	Вещественный
	Абсцисса 1	Вещественный
	Ордината 1	Вещественный

	Абсцисса N	Вещественный
Выходы	Ордината N	Вещественный
	Выход	Вещественный

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

Регулятор

В зависимости от того, работает алгоритм регулирования в контроллере или в MasterSCADA, состав входов и выходов ФБ может различаться. Вид функционального блока в дереве объекта (левый рисунок – Регулятор в MasterSCADA, правый – в контроллере):



Входы и выходы:

	Имя	Тип	Примечание
Входы	Параметр	Вещественное дв. точности	Регулируемый параметр (PV).
	Задание	Вещественное дв. точности	Задание (SP).
	Открыт	Логический	Конечный выключатель открытого положения.
	Закрыт	Логический	Конечный выключатель закрытого положения.
	Снять аварию	Логический	Если Истина, в окне оператора прячется кнопка <i>Квитировать</i> , клапан перестает мигать.
	Неисправность питания	Логический	Зажигает лампочку в окне оператора, сигнализирующую о неисправности питания.
	Блокировка	Логический	Управление включением режима <i>Блокировки</i> , выход регулятора <i>Управление</i> получает значение, определенное в <i>Настройках</i> (открыт/закрыт/остановлен).
	Положение	Вещественное дв. точности	Сигнал от указателя положения исполнительного механизма (0-100%).
	Режим программа	Логический	Включает режим <i>Программа</i> (управление выходом регулятора от внешнего алгоритма). Если алгоритм регулирования работает в контроллере, этот <i>Вход</i> не показывается.
	Управление от программы	Вещественное дв. точности	В режиме <i>Программа</i> значение этого входа передается на выход <i>Управление</i> . Предварительно нормируется (приводится к шкале 0-100%) и ограничивается скорость изменения. В остальных режимах не используется.
	Группа входов <i>Настройки</i>		Если алгоритм регулирования работает в контроллере, эта группа входов не показывается. Входы предназначены для предварительного задания настроечных констант, либо получения значения от внешнего алгоритма расчета настроек регулятора. Значения входов группы используются в алгоритме регулятора только по переднему фронту входа <i>Обновить</i> . Для использования проектных констант достаточно, чтобы на входе <i>Обновить</i> при разработке была задана константа 1.
	К пропорц.	Вещественное дв. точности	Коэффициент пропорциональной части.
	Выходы	Т интегр.	Вещественное дв. точности
Т диффер.		Вещественное дв. точности	Постоянная времени дифференцирования <i>Т диффер</i> . Если равен 0, дифференциальная часть отключается.
Обновить коэффициенты.		Логический	Только по переднему фронту этого входа считываются значения входов <i>К пропорц.</i> , <i>Т интегр.</i> , <i>Т диффер</i> .
Выход <i>Задание</i>		Вещественное дв. точности	В зависимости от режима может быть равно входу <i>Задание</i> или значению, заданному оператором в окне управления ФБ
Управление		Вещественное дв. точности	Выход регулятора (управляющее воздействие на исполнительный механизм). Формируется в диапазоне от 0 до 100%. Для нормально-открытого клапана (настройка <i>Тип клапана</i> <i>НО</i>) выход пересчитывается по формуле (100% - <i>Управление</i>). Скорость изменения выхода ограничивается. Перед выдачей на выход ФБ значение управляющего сигнала пересчитывается из диапазона 0-100% в диапазон изменения, указанный для этого выхода.
Дистанционный режим включен		Логический	Значение режима <i>Дистанционное управление/Автомат</i> .
Авария		Логический	См. пункт Формирование аварии.

Имя	Тип	Примечание
Группа выходов <i>Настройки</i>		Эта группа выходов показывается, если алгоритм регулирования работает в контроллере. Выходы предназначены для настройки работающего в контроллере регулятора. Передаются заданные выходам в режиме разработки константы, либо введенные оператором в окне управления значения.
К пропорц.	Вещественное дв. точности	Коэффициент пропорциональной части формулы регулятора.
Т интегр.	Вещественное дв. точности	Постоянная времени интегрирования <i>T интегр.</i>
Т диффер.	Вещественное дв. точности	Постоянная времени дифференцирования <i>T диффер.</i>

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Существует 4 режима работы ПИД-регулятора:

- Программное управление (ПУ);
- Каскад;
- Автомат;
- Дистанционное управление.

Кроме того регулятор может находиться в состоянии *Блокировка*.

Режим Программное управление (ПУ) используется в случае, если формирование управляющего воздействия на исполнительный механизм осуществляется внешним алгоритмом. В режиме ПУ данные со входа *Управление* от программы передаются непосредственно на выход ФБ *Управление*. При этом оператор в окне управления может изменять выход *Задание*, например, для использования во внешнем алгоритме, осуществляющем регулирование, или для подготовки перехода в режим *Автомат* или для переключения регулятора из режима работы в качестве задающего контура каскадной схемы в режим прямого управления ведомым контуром.

Окно управления в режиме ПУ:

- Слайдер *Задание* и поле ввода *Задание* отображают последнее полученное ранее значение и доступны для воздействия;
- Слайдер *Управление* и поле ввода *Управление* отображают значение входа *Управление* от программы и недоступны для воздействия.

Режим Каскад используется при работе регулятора в качестве ведомого контура в каскадной схеме. В этом режиме *Задание* формируется не оператором, а берется со входа *Задание*. Управляющее воздействие на исполнительный механизм рассчитывается с помощью внутреннего алгоритма регулирования.

Окно управления в режиме *Каскад*:

- Слайдер *Задание* и поле ввода *Задание* отображают значение входа *Задание* и недоступны для воздействия;
- Слайдер *Управление* и поле ввода *Управление* отображают формируемое в ФБ значение и недоступны для воздействия.

Режим Автомат - основной режим работы *Регулятора*. В этом режиме управляющее воздействие формируется Функциональным блоком, с помощью внутреннего алгоритма регулирования. Значение *Задания* формируется оператором из окна управления.

Окно управления в режиме *Автомат*:

- Слайдер *Задание* и поле ввода *Задание* отображают последнее полученное ранее значение и доступны для воздействия;
- Слайдер *Управление* и поле ввода *Управление* отображают формируемое в ФБ значение и недоступны для воздействия.

Режим Дистанционное управление (ДУ) - в этом режиме алгоритм регулирования функционального блока не используется. Управляющее воздействие на исполнительный механизм формируется оператором из окна управления.

Окно управления в режиме Дистанционный:

- Слайдер *Задание* и поле ввода *Задание* отображают последнее полученное ранее значение и доступны для воздействия.

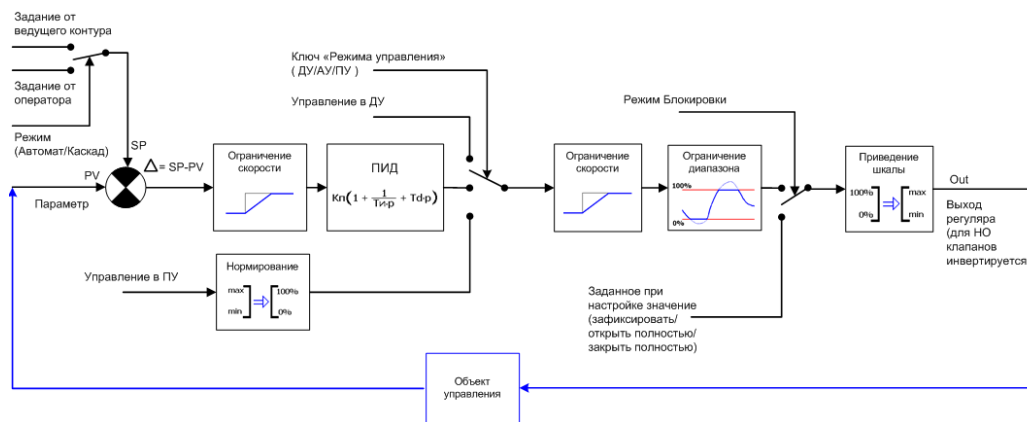
- Слайдер *Управление* и поле ввода *Управление* отображают последнее полученное ранее значение и доступны для воздействия.

Состояние Блокировка. Регулятор переключается в это состояние при наличии сигнала *Истина* на входе *Блокировка*. В окне управления все кнопки переключения режима становятся недоступны, на их месте включается надпись *Блокировка* красного цвета. При переходе в блокировку выполняется действие, заданное на странице свойств странице свойств *Настройки*. Возможно три варианта: сигнал *Управление* не изменяется, Сигнал *Управление* получает значение 100%, Сигнал *Управление* получает значение 0%.

Окно управления в режиме *Блокировка*:

- Слайдер *Управление* и поле ввода *Управление* отображают заданное значение и недоступны для воздействия;
- Слайдер *Задание* и поле ввода *Задание* отображают последнее полученное ранее значение и доступны для воздействия.

Формирование управляющего воздействия производится только в режимах *Каскад* и *Автомат*. Функциональная схема работы ПИД-регулятора аналогового показана на рисунке ниже.



При старте MasterSCADA режим работы регулятора определяется в два этапа - определение доступных режимов и выбор исходного режима.

Если регулятор расположен в MasterSCADA возможна работа в режимах:

- *Режим ПУ* доступен, если вход *Режим Программа* имеет подключенную связь, в противном случае этот режим считается недоступным, и кнопка включения этого режима в окне оператора отсутствует;
- *Режим Каскад* доступен только в том случае, если вход *Задание* имеет подключенную связь, в противном случае этот режим недоступен, и кнопка включения этого режима в окне оператора отсутствует;
- *Режим Автомат* доступен всегда;
- *Режим ДУ* доступен всегда.

Если регулятор расположен в контроллере возможна работа только в режимах *Автомат* и *ДУ*, причем эти режимы доступны всегда.

Выбор исходного режима работы зависит от настроек *Старт в автоматическом режиме* и *Восстановление режима при рестарте*, задаваемых на странице свойств функционального блока *Настройки*.

Выбор исходного режима если регулятор расположен в MasterSCADA:

- *Восстановление режима при рестарте* - если флаг установлен, то при последующем рестарте MasterSCADA будет восстановлен режим регулятора, действовавший в момент остановки режима исполнения в предыдущей сессии;
- *Старт в автоматическом режиме* - если флаг не установлен, то включается *Дистанционный режим*. Если же этот флаг установлен, то включится один из трех автоматических режимов - *ПУ*, *Каскад* или *Автомат* в зависимости от состояния входов ФБ:
 - Режим *ПУ* включается, если он доступен, и на входе *Режим Программа* присутствует сигнал *Истина*;
 - Если не включился режим *ПУ*, то включается режим *Каскад*, при условии, что он доступен, иначе включается режим *Автомат*.

Выбор исходного режима если регулятор расположен в контроллере:

- *Восстановление режима при рестарте* - если флаг установлен, то при последующем рестарте MasterSCADA будет восстановлен режим регулятора, действовавший в момент остановки режима исполнения в предыдущей сессии;
- *Старт в автоматическом режиме* - если флаг установлен, то включается режим *Автомат*, иначе включается режим *ДУ*.

Во время работы MasterSCADA переключение режимов осуществляется из окна управления. Кроме того, для включения режима *ПУ* можно использовать вход *Режим Программа*.

Существует 3 условия формирования аварии: *Авария по неоткрытию клапана*, *Авария по незакрытию* и *Невыполнение команды*.

Авария по неоткрытию формируется, если по прошествии интервала времени *Контрольное время* не выполняется условие: $(\text{Выход Управление} = 100\%) \text{ И } (\text{Включен вход Выход Открыт ИЛИ } (\text{Положение} > (100\% - \text{Допустимое отклонение})))$.

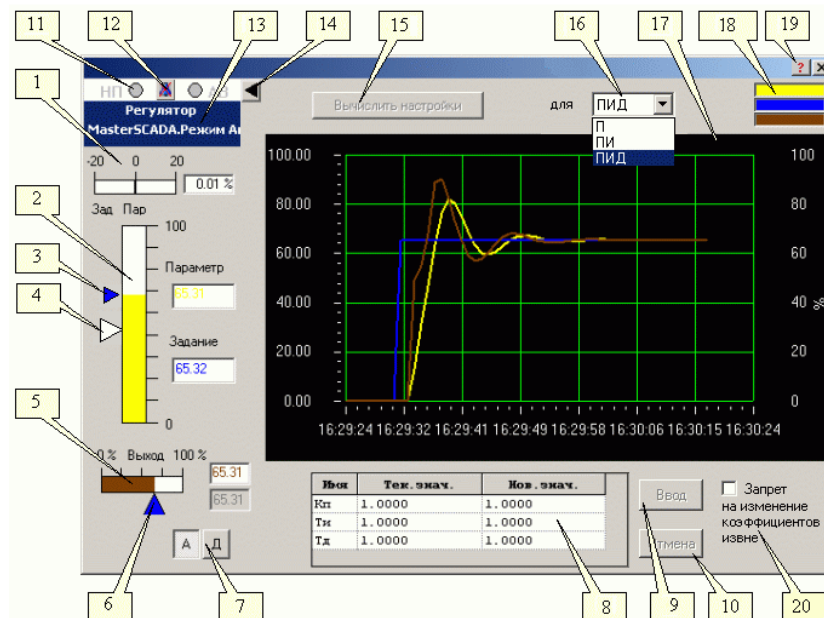
Авария по незакрытию формируется, если по прошествии интервала времени *Контрольное время* не выполняется условие: $(\text{Выход Управление} = 0\%) \text{ И } (\text{Включен вход Закрыт ИЛИ } (\text{Выход Положение} > \text{Допустимое отклонение}))$.

Невыполнение команды формируется, если по прошествии интервала времени *Контрольное время* не выполняется условие: $|\text{Выход Управление} - \text{Положение}| < \text{Допустимое отклонение}$.

Авария формируется при срабатывании хотя бы одного из условий, отмеченного флажком на странице свойств ФБ *Настройка*. Настройка параметров *Контрольное время* и *Допустимое отклонение* осуществляется там же.

- ❖ **Важно!** Если какой-либо из входов *Открыт*, *Закрыт* и *Положение* не используется, то условие с его использованием при формировании аварии не проверяется.

При нажатии в режиме исполнения левой кнопки мыши на изображении ВФБ в мнемосхеме открывается *Окно управления*, вид которого показан на рисунке ниже.



Описание элементов управления:

- *Индикатор Рассогласование (1)* - отображает значение рассогласования (разность между *Заданием* и *Параметром*);
- *Индикатор Параметр (2)* - отображает текущее значение *Параметра*;
- *Индикатор Задание (3)* - отображает текущее значение *Задания*;
- *Слайдер Задание (4)* - в режимах *ПУ*, *Автомат* и *ДУ* позволяет устанавливать значение *Задания*;
- *Индикатор Положение (5)* - отображает текущие значения положения клапана;
- *Слайдер Управление (6)* - в режиме *ДУ* позволяет устанавливать значение выхода *Управления*;

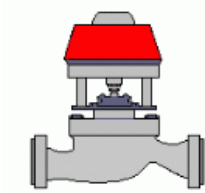
- *Кнопки переключения режимов (7)* - показываются доступные режимы. В режиме исполнения с помощью этих кнопок осуществляется переключение режимов;
- *Таблица ввода коэффициентов (8)* - в режиме исполнения позволяет вводить новые коэффициенты регулятора. В столбце *Текущие значения* выводятся действующие значения коэффициентов. Новые значения вводятся в ячейках столбца *Новые значения*;
- *Кнопка Ввод (9)* - по нажатию кнопки *Ввод* происходит ввод новых коэффициентов в формулу алгоритма ПИДа. Эта кнопка недоступна, если в таблице коэффициентов не были введены новые значения;
- *Кнопка Отмена (10)* - при нажатии на эту кнопку отменяется ввод новых коэффициентов в таблице. При этом в столбце *Новые значения* таблицы устанавливаются текущие коэффициенты;
- *Индикаторы Неисправность питания и Авария (11)* - Сигнализируют о *Неисправности питания* и *Аварии* соответственно. В неквитированном состоянии индикаторы мигают красным цветом. После квитирования индикаторы перестают мигать и горят красным цветом. После того как условия, вызвавшие появление состояний *Авария* и *Неисправность питания* перестают быть актуальными (на входе *Неисправность питания* устанавливается *Ложь* и функциональный блок перестает выдавать сигнал *Авария*), индикаторы гаснут;
- *Кнопка Квитировать (12)*; - квитирование *Неисправности питания* и *Аварии*;
- *Заголовок окна управления (13)* - выводимый в заголовке текст задается на странице свойств элемента *Вид*;
- *Кнопка Открыть/Закрыть тренд (14)* - с помощью этой кнопки, можно расширять окно управления зоной отображения тренда и таблицы ввода коэффициентов алгоритма ПИДа;
- *Кнопка Вычислить настройки (15)* - активизирует функцию автоматического вычисления коэффициентов регулятора. При отсутствии опции настройки коэффициентов регулятора в комплекте поставки функция не вызывается;
- В этом списке (16) можно выбрать тип регулятора, который планируется использовать после автоматического вычисления коэффициентов регулятора;
- *Тренд (17)* - позволяет в режиме исполнения просматривать значения *Параметра*, *Задания* и *Положения* в виде графика;
- *Кнопки включения и отключения перьев тренда (18)* - *Параметра*, *Задания* и *Положения*. Цвет перьев задается на странице свойств элемента *Цвета* при настройке;
- *Кнопка открытия окна Недостовверные данные (19)* - эта кнопка появляется только в том случае, если на входы ФБ приходят недостоверные данные. При нажатии на эту кнопку открывается диалоговое окно, показывающее данные каких входов недостоверны;
- *Флаг Запрет на изменение коэффициентов извне (20)* - установленный флаг запрещает использовать коэффициенты, полученные со входов ФБ (*К пропорц.*, *T интегр.*, *T диффер.*).

Для создания элемента управления в мнемосхеме, связанного с функциональным блоком *ПИД-регулятор аналоговый*, достаточно просто перетащить ФБ из дерева объектов в мнемосхему. Изображение ВФБ ПИД-регулятор аналоговый в мнемосхеме выглядит следующим образом:


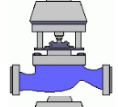



Изображение функционального блока имеет две сигнальные области, информирующих оператора о наличии аварии, а также о положении органа управления исполнительного механизма. Кроме того, изображение ВФБ в мнемосхеме сигнализирует о недостоверности входных данных и о пребывании ФБ в состоянии *Блокировка*.

В состоянии неквитированной аварии, аварийная сигнальная область начинает мигать красным цветом. После квитирования мигание снимается:



Положение органа управления определяется функциональным блоком в зависимости от данных на входах *Открыт*, *Закрыт* и *Положение*, и настройки *Допустимое отклонение*, задаваемой на странице свойств *Настройки*, в соответствии со следующей таблицей:

Состояние	Условие	Изображение
Полностью открыт	При наличии входа <i>Положение</i> - 100% - <i>Положение</i> < <i>Допустимое отклонение</i> . При отсутствии входа <i>Положение</i> и наличии входа <i>Открыт</i> - <i>Вход Открыт</i> = <i>Истина</i> .	
Промежуточное состояние	При наличии входа <i>Положение</i> - <i>Допустимое отклонение</i> < <i>Положение</i> < 100% - <i>Допустимое отклонение</i> . При отсутствии входа <i>Положение</i> и наличии входа <i>Открыт</i> - <i>Вход Открыт</i> = ЛОЖЬ и <i>Вход Закрыт</i> = ЛОЖЬ.	
Полностью закрыт	При наличии входа <i>Положение</i> - <i>Положение</i> < 0% + <i>Допустимое отклонение</i> . При отсутствии входа <i>Положение</i> и наличии входа <i>Закрыт</i> - <i>Вход Закрыт</i> = <i>Истина</i> .	

- ❖ **Важно!** *Вход Положение имеет более высокий приоритет по отношению ко входам Открыт и Закрыт, поэтому, если вход Положение используется, то определение положения органа управления исполнительного механизма осуществляется исходя из данных на этом входе, вне зависимости от состояния входов конечных выключателей Открыт и Закрыт. При отсутствии входа Положение, положение органа управления определяется состоянием входов Открыт и Закрыт.*


Цвет сигнальной области для состояния *Полностью открыт* соответствует настройке *Основной цвет потока*, задаваемой на странице свойств *Цвет*. Для промежуточного состояния цвет потока определяется автоматически (становится более светлым), исходя из цвета, заданного для состояния *Полностью открыт*. Для состояния *Полностью закрыт* цвет сигнальной области серый.

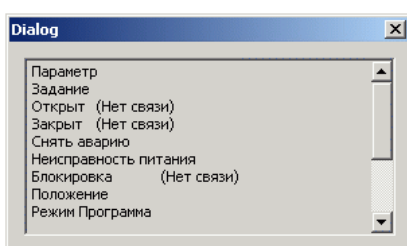
В режиме блокировки текущее изображение дополняется черным прямоугольником:



При наличии признака недостоверности данных входов, изображение ФБ в мнемосхеме становится более бледным:



При наличии признака недостоверности данных входов, в заголовке окна управления появляется кнопка , при нажатии на которую открывается диалоговое окно, содержащее детализацию информации о недостоверности.



Данные, пришедшие на входы функционального блока считаются недостоверными, если хотя бы один из входов функционального блока имеет *Плохой* ОРС признак качества. В диалоговом окне недостоверности данных отображается полный список входов функционального блока. Для входов которые имеют *Плохой* ОРС-признак качества, приводится его расшифровка.

Выполняется хотя бы одно из следующих условий:

- Включен вход *Закрит И (Положение > Допустимое отклонение)*, в окне недостоверности этому состоянию соответствует строка *Входы Закрит и Положение не совпадают*;
- Включен вход *Открыт И (Положение < (100% - Допустимое отклонение))*, в окне недостоверности этому состоянию соответствует строка *Входы Открыт и Положение не совпадают*.

Допустимое отклонение - настройка, задаваемая на странице свойств Настройки.

ФБ ПИД-регулятор аналоговый способен формировать следующие сообщения:

Сообщение	Условие	Категория	Формирование по умолчанию
Изменено управление	Оператор изменил значение <i>Управления</i>	Информация	нет
Изменено задание	Оператор изменил значение <i>Задания</i>	Информация	нет
Изменены коэффициенты	Оператор изменил значение коэффициентов	Информация	нет
Изменён запрет обновления коэффициентов от входов	В окне управления оператор изменил состояние флага, разрешающего установку коэффициентов от входов функционального блока	Информация	нет
Изменён режим	Оператором был изменен режим работы регулятора	Информация	нет
Квитирование	Оператором нажата кнопка <i>Квитировать</i>	Информация	нет
Авария	Функциональным блоком был сформирован сигнал <i>Авария</i>	Авария	есть
Неисправность питания	На входе функционального блока неисправность питания был установлен сигнал Истина	Авария	есть

Перечень допустимых действий операторов в окне управления ВФБ *ПИД-регулятор* аналоговый (как и для любого другого исполнительного механизма) настраивается на закладке *Операторы* страниц свойств системы в категории *Исполнительный механизм*. Детализация обобщенных названий действий на этой закладке для ВФБ *ПИД-регулятор* приведена ниже.

Действие	Описание
Изменить режим	Изменение оператором режима работы регулятора из окна управления.
Изменить величину	Установка оператором значения <i>Задания</i> или <i>Управления</i> из окна управления.
Квитирование	Квитирование оператором <i>Аварии</i> и <i>Неисправности питания</i>
Изменить настройки	Ввод оператором новых значений коэффициентов в окне управления.
Открыть окно управления	Открытие оператором окна управления

В ФБ *ПИД-регулятор* аналоговый реализована возможность автоматического определения оптимальных коэффициентов регулятора. Чтобы активизировать функцию автоматического вычисления коэффициентов регулятора, необходимо проделать следующие действия:

- Открыть окно управления регулятора, щелкнув левой кнопкой мыши по изображению регулятора в мнемосхеме, и развернуть дополнительное поле, содержащее график и таблицу ввода коэффициентов;
- Изменить задание;
- Дождаться окончания переходного процесса. Этот момент можно определить визуально с помощью графика;
- Выбрать тип регулятора, который предполагается использовать;
- Нажать кнопку *Вычислить настройки*. Рассчитанные коэффициенты автоматически установятся в таблице коэффициентов и вступят в силу только после подтверждения оператором (по кнопке *Ввод*).

Страница свойств *Настройки* ФБ ПИД-регулятор аналоговый выглядит следующим образом:

Группа *Действия при блокировке*. При переходе в состояние *Блокировка* выполняется выбранное действие:

- *Зафиксировать* – заморозить последнее перед переключением в состояние *Блокировка* значение выхода *Управление*;
- *Закреть полностью* - установить выход *Управление* = 0%;
- *Открыть полностью* - установить выход *Управление* = 100%.

Группа *Максимальная скорость изменения Рассогласования и Управления*. Механизм балансировки не позволяет изменяться этим параметрам со скоростью, больше указанной. Если максимальная скорость изменения *Рассогласования* или *Управления* равна 0, то ограничения скорости изменения не происходит.

Группа *формирование аварии*. *Авария* может формироваться по нескольким условиям одновременно:

- Авария по неоткрытию;
- Авария по незакрытию;
- Невыполнение команды.

Флаги *Старт в автоматическом режиме* и *Восстановление режима при рестарте* определяют выбор режима работы регулятора при старте MasterSCADA.

Страница свойств *Состав* представлена на рисунке ниже.

С помощью переключателей группы *Конечники* задается наличие входов функционального блока, принимающих сигналы от конечных выключателей (*Входы Открыт* и *Закрыт*):

- *Оба* - используются оба входа *Открыт* и *Закрыт*;
- *Только открыт* - используется только вход *Открыт*;
- *Только закрыт* - используется только вход *Закрыт*;
- *Ни одного* - входы *Открыт* и *Закрыт* не используются.

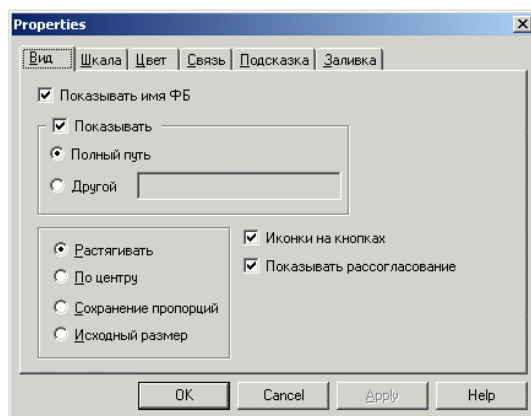
Если флаг *Использовать вход Положение* не установлен, вход *Положение* отсутствует. Группа *Регулятор* позволяет задать *Прямой* или *Обратный* характер управления регулятором:

- *Прямой* - значение рассогласования формируется как разность между величиной *Задания* и текущим значением регулируемого параметра;
- *Обратный* - регулятор работает с инверсией рассогласования, которое рассчитывается в этом режиме как разность между регулируемым параметром и *Заданием*.

При *Прямом* управлении, увеличение значения на входе *Параметр* приводит к увеличению выхода ФБ *Управление*, а при *Обратном* к уменьшению. Если в группе *Тип клапана* уста-

новлено *НО* (Нормально открытый), то значение выхода инвертируется, т.е. перед выдачей значения управления на выход, функциональный блок производит его пересчет по формуле ($100\% - \text{Управление}$).

На закладке *Вид* настраивается внешний вид окна управления, а также изображения регулятора в мнемосхеме.



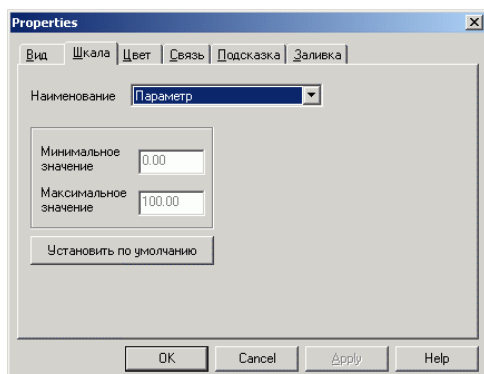
Функции элементов управления:

- Флаг *Показывать имя ФБ*, заголовок окна управления состоит из двух строк. В верхней может выводиться имя функционального блока в дереве объектов. В нижней может отображаться либо полный путь к функциональному блоку, либо произвольный текст. Установленный флаг *Показывать имя ФБ* означает, что в верхней строке заголовка окна управления будет отображаться имя функционального блока. При сброшенном флаге верхняя строка заголовка остается пустой;
- Флаг *Показывать* относится к нижней строке в заголовке окна управления. Если флаг установлен, то в этой строке будет выводиться либо полный путь к функциональному блоку в дереве объектов, либо произвольный текст. Если флаг сброшен, нижняя строка заголовка остается пустой;
- В случае, если флаг *Иконки на кнопках* установлен, на кнопках переключения режимов в окне управления отображаются иконки;
- Флаг *Показывать рассогласование* устанавливает, отображать или нет индикатор *Рассогласование* в окне управления.

С помощью группы переключателей устанавливается способ вывода изображения регулирующего клапана на мнемосхему:

- Растягивать;
- По центру;
- Сохранение пропорций;
- Исходный размер.

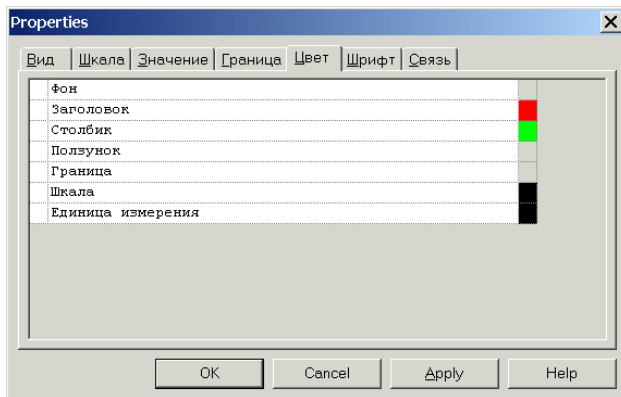
Вид закладки *Шкала* представлен на рисунке ниже.



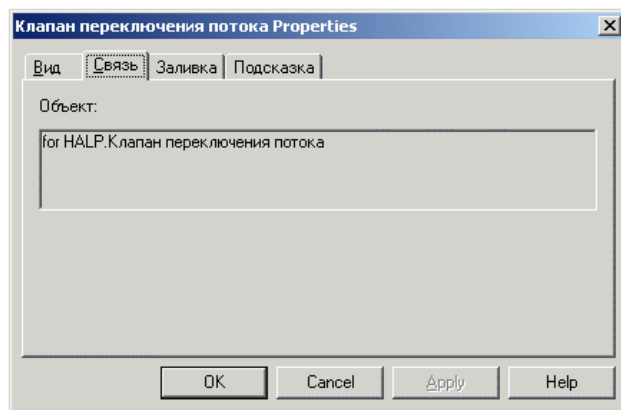
Выпадающий список *Наименование* позволяет выбрать, для какого элемента будет осуществляться настройка шкалы. Возможные значения:

- *Параметр* - настройки шкалы относятся к индикатору *Параметр* в окне управления;
- *Рассогласование* - настройки шкалы относятся к индикатору *Рассогласование* в окне управления.

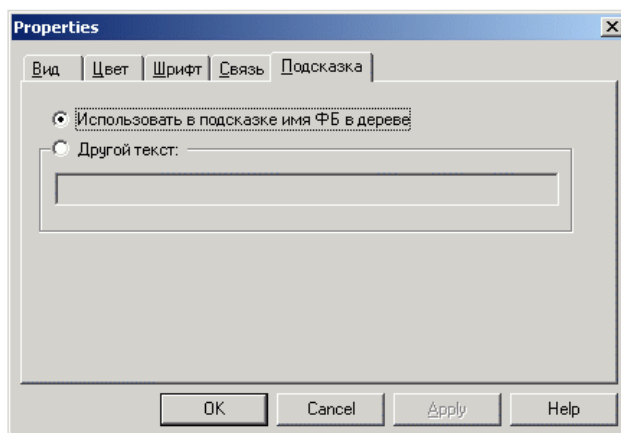
На закладке *Цвет* настраиваются цвета всех элементов ВФБ, отображаемых на мнемосхеме (заголовок, граница, шкала, значение в норме, ползунок и т.д.). Первоначально все элементы имеют свой стандартный цвет. Если необходимо сменить стандартный цвет, возможно его изменить. Для этого щелкните мышкой по цветному квадратику напротив названия элемента, чей цвет необходимо изменить. При этом появится стандартный диалог *Windows* для выбора цвета.



На закладке *Связь* отображается связь ВФБ с переменной проекта и путь до этой переменной в дереве.



На закладке *Подсказка* настраивается всплывающая подсказка, которая будет отображаться как в режиме разработки, так и в режиме исполнения при наведении курсора мыши на данный элемент. Подсказка может отображать полный путь к элементу в дереве объектов, или произвольный текст, введенный пользователем.



Закладка *Заливка* аналогична другим функциональным блокам, описанным в предыдущих разделах.

Таблица

ВФБ *Таблица* относится к категории *Управление* и имеет группу входов и группу выходов. Вид функционального блока в дереве объекта:



Таблица предназначена для отображения значений переменных в виде таблицы параметров. На поле *Таблицы* отображается номер параметра, его название, значение и единица измерения. Имена вновь добавляемых параметров индексируются по последнему имени.

Таблица ТЭП			
1	Кол-во воды	478.65	м3
2	Кол-во соли	154.21	м3
3	Кол-во щелочи	89.54	м3
4	Кол-во кислоты	245.87	м3

Единица измерения, формат значения и аварийные зоны наследуются от настроек переменных привязанных ко входам *Таблицы* в дереве объекта. Число входов и выходов *Таблицы*, их свойства настраиваются в проекте на странице свойств *Параметры*. Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

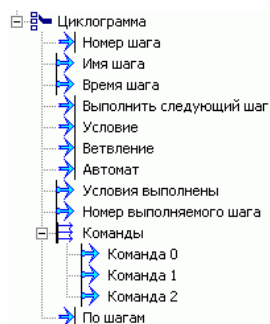
- ❖ **Важно!** Значение, отображаемое в *Таблице* - это значение входа функционального блока, а введенные данные подаются на выход. Поэтому, если нужно, чтобы в *Таблице* отображались введенные данные, следует соединить вход ФБ с соответствующим выходом.

Циклограмма

Функциональный блок *Циклограмма* предназначен для управления группой исполнительных механизмов. Состояния системы описываются последовательностью шагов, для каждого из которых можно настроить комбинацию команд. Количество шагов и команд настраивается на странице *Свойства*.

В режиме исполнения *Циклограмма* выполняет необходимую последовательность шагов, при этом при переходе на новый шаг на управляющих выходах *Циклограммы* появляется predetermined для этого шага набор команд. Команды настраиваются на странице свойств *Шаги*. Момент перехода на следующий шаг определяется по времени (задается минимальное время выполнения шага), и по внешнему условию. Кроме того, существует возможность прекратить выполнение текущего шага вне зависимости от истинности условий его окончания, и перейти на следующий.

Вид функционального блока в дереве объекта:



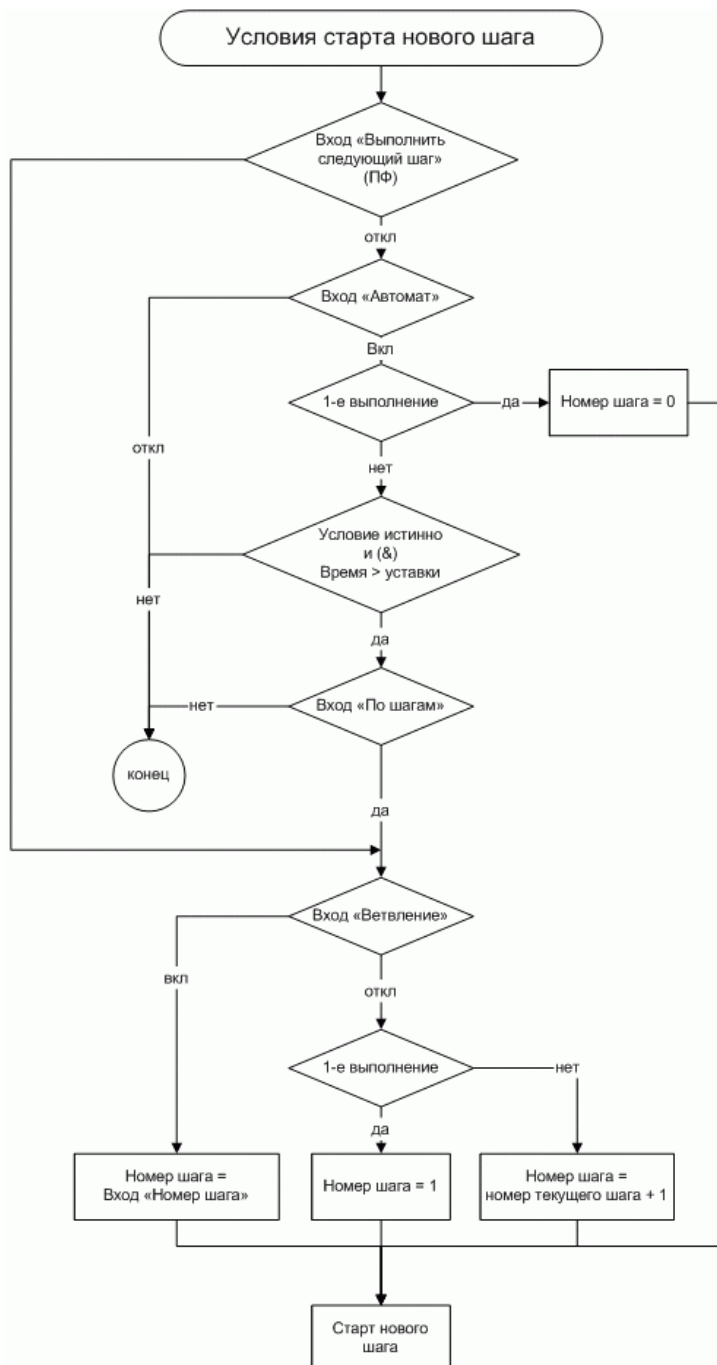
Входы и выходы:

Название	Тип	Примечание
Вход <i>Номер шага</i>	целый	Номер следующего шага. Используется только в том случае, если вход <i>Ветвление</i> = <i>Вкл.</i>
Вход <i>Выполнить следующий шаг</i>	логический	По переднему фронту сигнала происходит переход к следующему шагу. Выполнение текущего шага прекращается.
Вход <i>Условие</i>	вещественный	Служит для задания уставки для проверки условия окончания текущего шага.
Вход <i>Ветвление</i>	логический (по переднему фронту)	Определяет, как образом вычисляется номер следующего шага - если <i>Ветвление</i> = <i>Вкл.</i> , номер следующего шага берется со входа <i>Номер шага</i> . Если <i>Ветвление</i> = <i>Выкл.</i> , номер следующего шага рассчитывается как <i>номер текущего шага + 1</i> .
Вход <i>Автомат</i>	логический (по переднему фронту)	Если на входе <i>Автомат</i> установлено <i>Выкл.</i> , шаг не меняется, если <i>Вкл.</i> , проверяются условия окончания текущего шага.

Название	Тип	Примечание
Выход <i>Имя шага</i>	строка	Имя текущего шага, которое для каждого шага задается на странице свойств <i>Шаги</i> .
Выход <i>Время шага</i>	время	Разница между текущим временем и временем последнего изменения номера шага.
Выход <i>Условия выполнены</i>	логический	Показывает, были ли выполнены условия (если они проверялись), независимо от того, был ли старт нового шага.
Выход <i>Номер выполняемого шага</i>	целый	Номер текущего шага.
Группа <i>Выходов Команды</i>	логический	Значения команд, определяемые текущим шагом. Команды для каждого шага настраиваются на странице <i>Шаги</i> .
Вход <i>По шагам</i>	логический	Если на входе <i>По шагам</i> установлено <i>Вкл.</i> , шаг не меняется, если <i>Вкл.</i> , проверяются условия окончания текущего шага.

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Выбор следующего шага осуществляется функциональным блоком в соответствии с приведенной ниже блок-схемой:



Пояснения к схеме - *Вход Выполнить следующий шаг* является импульсным - он реагирует на передний фронт сигнала. При появлении переднего фронта на этом входе проверка

условий, а также времени выполнения текущего шага не производится, и *Циклограмма* сразу переходит к следующему шагу.

На странице *Свойств* устанавливается количество шагов и команд *Циклограммы*.

Вид страницы:

Свойство	Тип свойства	Значение
Количество шагов	Целый	8
Количество команд	Целый	8

На странице *Шаги* настраиваются состояния команд для каждого из шагов. Количество команд и шагов задается на странице *Свойства*.

Вид страницы:

Шаг	Имя шага	0	1	2
0	Исходное состояние	Откл	Откл	Откл
1	Открытие вентиля пара	Вкл	Откл	Откл
2	Включение компрессора	Вкл	Вкл	Нет
3	Включение подачи воды	Откл	Вкл	Вкл
4	Отключение	Откл	Откл	Откл

Каждая строка из вышеприведенной таблицы соответствует шагу - набору команд с определенными значениями. Для каждого шага можно задать имя, которое в режиме исполнения будет выводиться на выходе ФБ *Имя шага*. Состояния команд для каждого шага настраиваются в столбцах $0...N$, (N - количество команд). Изменение команды осуществляется щелчком левой кнопки мыши в соответствующей ячейке таблицы. Выходы команд могут принимать следующие значения:

- *Вкл.* - на выходе команды устанавливается сигнал *Истина*;
- *Откл.* - на выходе команды устанавливается сигнал *Ложь*;
- *Нет* - состояние выхода команды не меняется.

На странице *Условия* задаются условия окончания шагов, а также минимальное время их выполнения. Текущий шаг заканчивается, если действительное время выполнения шага больше заданного, и условие окончания шага истинно. Страница свойств *Условия* содержит единственную таблицу, которая показана ниже:

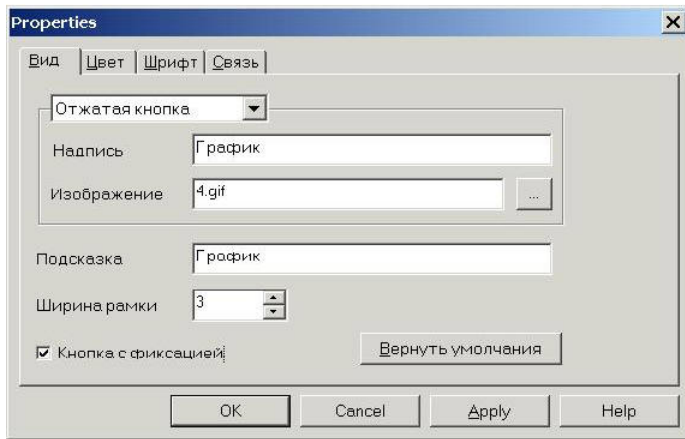
	Уставка времени, с	Уставка условия	Проверка условия
0	180	10	= (Равно)
1	90	0	Не проверять
2	90	0	Не проверять
3	20		ВКЛ
4	0		ВКЛ
5	3600	0	Не проверять
6	0		ОТКЛ
7	0	50	> (Больше)

Строки в этой таблице соответствуют шагам. Таблица состоит из следующих столбцов:

- *Уставка времени* - время в секундах, которое отводится на выполнение шага. Необходимо заметить, что в случае использования проверки окончания шага по условию и уставки времени на выполнение, реальное время шага может превышать заданное. Объясняется это тем, что ФБ *Циклограмма* считает, что шаг закончился, если и условие шага истинно, и время выполнения превысило заданное;
- *Уставка условия* - с этим значением сравнивается вход *Условие*;
- *Проверка условия* - в столбце выбирается оператор сравнения значения входа *Уставка* с уставкой условия.

Свойства Кнопки и Таблицы

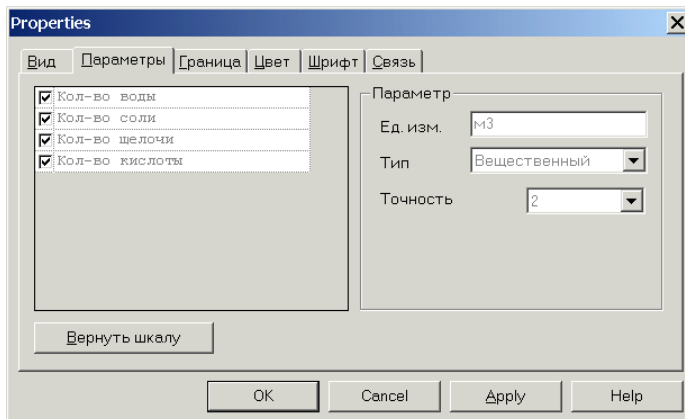
Страницы свойств Кнопки. На закладке *Вид* настраивается внешний вид ВФБ *Кнопка*.



Функции элементов управления описаны в таблице ниже:

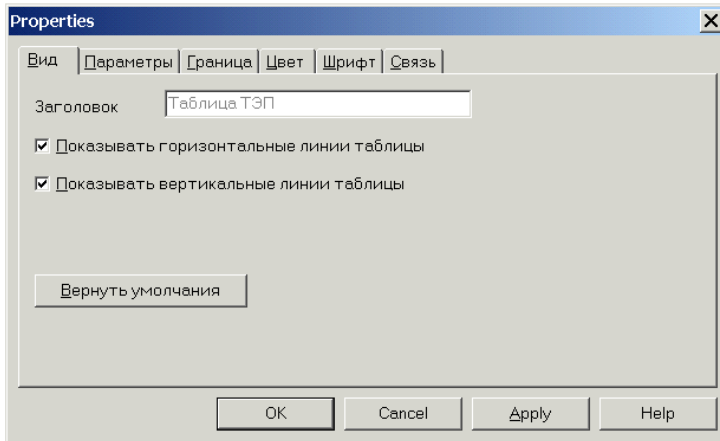
Элемент диалога	Описание
Поле выбора вида кнопки	Указывает для какого вида кнопки (отжатая или нажатая) будут вводиться последующие настройки.
Поле ввода <i>Надпись</i>	Наследуется от имени ВФБ в дереве объекта (название <i>Кнопка</i> в дереве объекта может быть изменено), новая надпись может быть задана в поле ввода, при отсутствии текста кнопка надписи иметь не будет.
Поле ввода <i>Изображение</i>	Указывает имя графического файла, изображение которого будет отображено на кнопке, при отсутствии текста кнопка рисунка иметь не будет.
Поле ввода <i>Подсказка</i>	Задаёт необходимость отображения подсказки и её содержание, при отсутствии текста подсказки не будет.
Поле ввода <i>Ширина рамки</i>	Задаёт значение ширины рамки кнопки.
Флажок <i>Кнопка с фиксацией</i>	Задаёт вид кнопки, как кнопки с фиксацией при нажатии.
Кнопка <i>Вернуть умолчания</i>	Восстанавливает свойства унаследованные из дерева объекта.

Страницы свойств Таблицы. На закладке *Параметры* задаются настройки отображения параметров ВФБ *Таблица*.



Функции элементов управления описаны в таблице ниже:

Элемент диалога	Описание
Список отображаемых параметров	Флажками задаются те параметры, которые должны отображаться в таблице, название параметра может быть изменено.
Поле ввода <i>Ед. изм.</i>	Единица измерения наследуется из свойств переменной дерева объекта и может быть изменена, при отсутствии текста единица измерения не пишется.
Поле ввода <i>Тип</i>	Тип значения наследуется из свойств переменной дерева объекта и может быть изменен, при этом будет производиться необходимое преобразование значения.
Выпадающий список <i>Точность</i>	Точность представления числа (число знаков после запятой) наследуется из свойств переменной дерева объекта и может быть изменена.
Кнопка <i>Вернуть шкалу</i>	Восстанавливает свойства, унаследованные из дерева объекта.
На закладке <i>Вид</i> настраивается внешний вид ВФБ <i>Таблица</i> .	

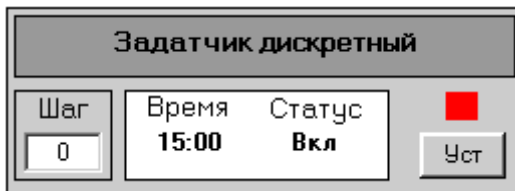


Функции элементов управления описаны в таблице ниже:

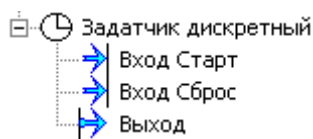
Элемент диалога	Описание
Поле ввода <i>Заголовок</i>	Имя наследуется от имени ВФБ в дереве объекта (название <i>Таблица</i> в дереве объекта может быть изменено), новое имя может быть задано в поле ввода, при отсутствии текста таблица заголовка не имеет.
Флажок <i>Показывать горизонтальные линии таблицы</i>	Задаёт отображение горизонтальных линий таблицы.
Флажок <i>Показывать вертикальные линии таблицы</i>	Задаёт отображение вертикальных линий таблицы.
Кнопка <i>Вернуть умолчания</i>	Восстанавливает свойства, унаследованные из дерева объекта.

Задатчик дискретный

Визуальный функциональный блок *Задатчик дискретный* предназначен для управления технологическими процессами, выполняемыми по расписанию.



Вид функционального блока в дереве *Объекта*:



Входы и выходы:

	Имя	Тип	Примечание
Входы	Старт	Логический	
	Сброс	Логический	
	Выход	Логический	Выход принимает состояние <i>Истина</i> или <i>Ложь</i> при совпадении установленного времени (в настройках таймера) с астрономическим временем.
Выходы			

Логика работы функционального блока *Задатчик дискретный* отражена в таблице:

Старт	Сброс	Статус таймера
Ложь	Ложь	Не запущен
Истина	Ложь	Запущен
Истина	Истина	На выходе 0
Ложь	Истина	На выходе 0

Страница свойств *Настройки* функционального блока *Задатчик дискретный* состоит из двух вкладок: *Параметры*, *Список шагов*.

Параметры | Список шагов

Часы: 10 | Минуты: 0

Номер шага: 0

Статус: Вкл. Выкл.

Период:

Ежедневно

Пн Вт Ср Чт Пт Сб Вс

Неделя 1

Неделя 2

Неделя 3

Неделя 4

Неделя 5

Период: 2008 Год

Ежегодно: 9 Месяц

Ежемесячно: 2 Число

Дата

Новый | Очистить

Рассмотрим подробнее вкладку *Параметры*.

С помощью переключателей группы *Задание шагов* задаются шаги, выполняемые по расписанию:

- *Часы/Минуты* - устанавливается время выполнения шага;
- *Номер шага* - задается порядковый номер шага;
- *Статус* - задается состояние Выхода на каждом из шагов;
- *Новый* - добавляет новый шаг.
- *Очистить* - удаляет выбранный шаг.

Часы: 10 | Минуты: 0

Номер шага: 0

Статус: Вкл. Выкл.

Новый | Очистить

С помощью переключателей группы *Период* устанавливается срок исполнения шагов:

- *Дата* - запрограммированные шаги будут исполняться в выбранный день;
- *Ежемесячно* - запрограммированные шаги будут исполняться в выбранный день каждого месяца;
- *Ежегодно* - запрограммированные шаги будут исполняться в выбранный день ежегодно;
- *Период* - шаги исполняются с выбранным периодом;
- *Ежедневно* - запрограммированные шаги будут исполняться каждые сутки. Если активен выбор *Ежедневно* и выбраны определённые дни, то шаги исполняются в выбранные дни;
- *Еженедельно* - запрограммированные шаги будут исполняться в выбранные дни выбранных недель.

Период

Ежедневно

Пн Вт Ср Чт Пт Сб Вс

Еженедельно

Неделя 1

Неделя 2

Неделя 3

Неделя 4

Неделя 5

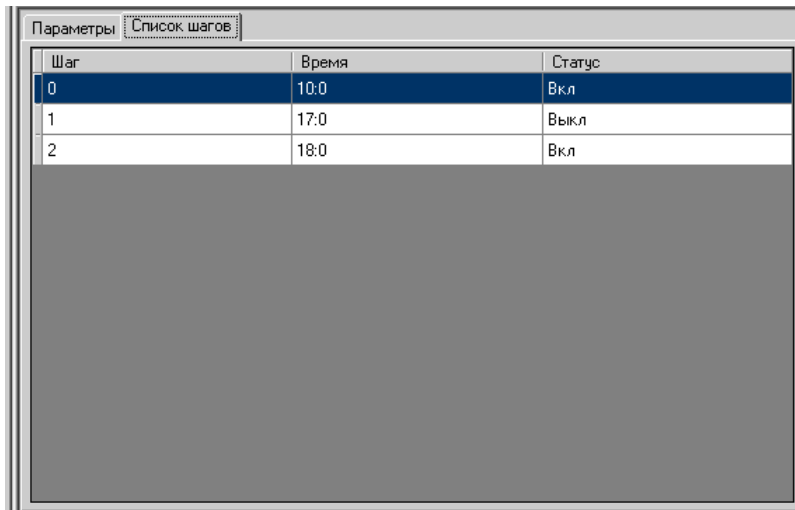
Период: 2008 Год

Ежегодно: 9 Месяц

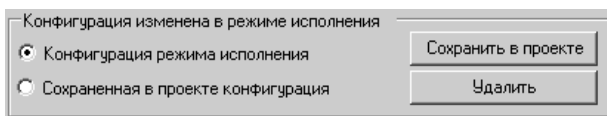
Ежемесячно: 2 Число

Дата

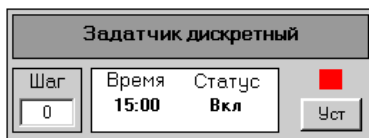
При нажатии на вкладку *Список шагов* открывается таблица в виде списка строк, где в каждой строчке указан номер шага, состояние, время включения и время выключения. В списке отображается запрограммированное количество шагов.



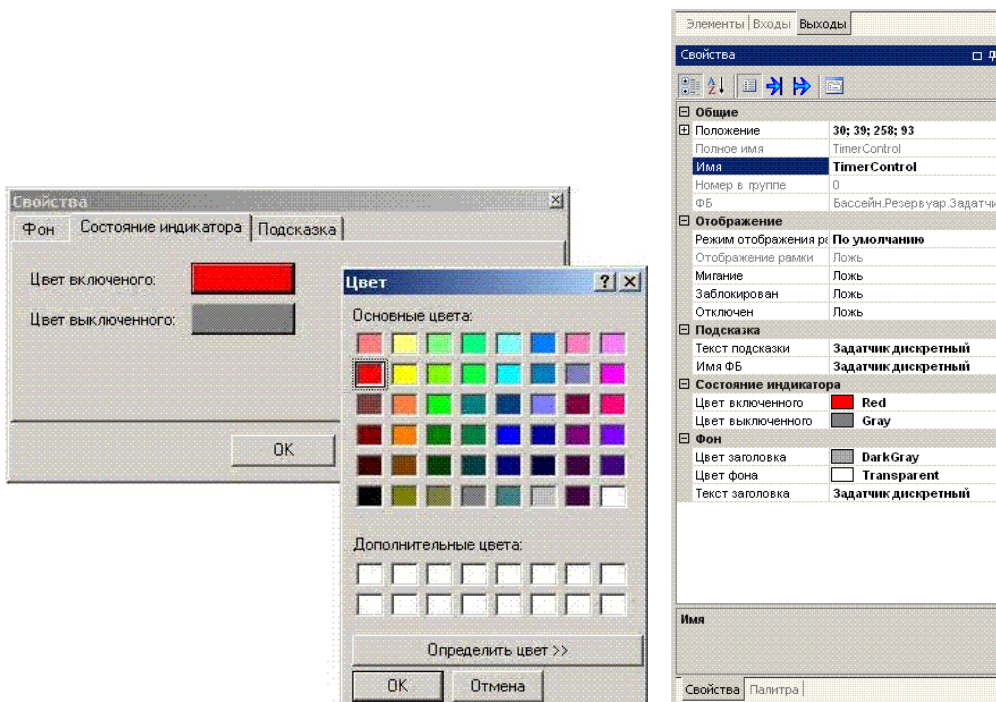
Панель *Конфигурация изменена в режиме исполнения* служит для сравнения/переключения настроек, назначенных в режиме исполнения либо сохраненных в режиме редактирования.



Внешний вид функционального блока *Задатчик дискретный* представлен на рисунке.

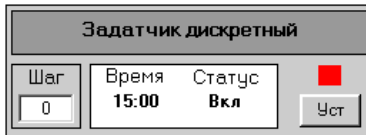


Цветовой индикатор ■ отображает состояние ^{Статус Вкл} *Выхода* функционального блока (Вкл./Выкл.). Цвет индикатора можно настраивать с помощью диалогового окна свойств, которое открывается при нажатии правой кнопкой мыши по функциональному блоку в режиме редактирования. Здесь настраивается внешний вид *Задатчика*, текст заголовка и подсказка.

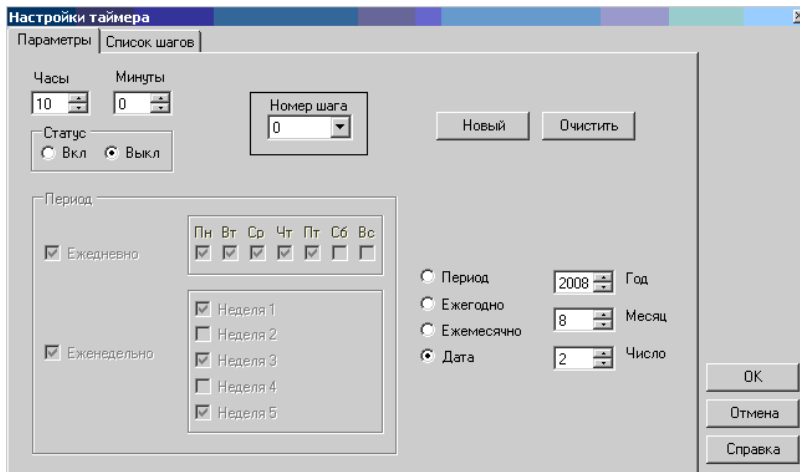


Эти и другие стандартные свойства ФБ настраиваются так же на Панели свойств функционального блока.

Окно управления дискретного задатчика выглядит следующим образом:



Кнопка **Уст** служит для задания времени включения, выключения технологического процесса, выполняемого по расписанию. Это время может быть задано, как в режиме разработки, так и в режиме исполнения. В режиме разработки это можно сделать на *Странице свойств Настройки* функционального блока. В режиме исполнения при нажатии на кнопку **Уст**. При нажатии в режиме исполнения на кнопку **Уст** открывается окно управления вкладка *Параметры*.



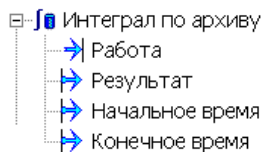
В нём задаётся время, номер шага, действие и нужное время совершения этого действия. Кнопка **OK** принимает выбранные настройки и активизирует работу таймера. Кнопка **Отмена** соответственно отменяет настройки.

Функциональны блоки для работы с архивом

Интеграл по архиву

ФБ *Интеграл по архиву* считает интеграл по выбранной переменной (команде, значению, расчету) за заданный период времени.

Вид функционального блока в дереве объекта:



Входы и выходы:

- *Работа* - признак включения ФБ в работу;
- *Результат* - полученное интегральное значение;
- *Начальное время* - фактическое время первого значения в архиве за заданный период времени;
- *Конечное время* - фактическое время последнего значения в архиве за заданный период времени.

Параметры настройки:

- *Имя параметра* - переменная, для которой вычисляется ФБ;
- *Начало* - время, начиная с которого производится подсчет;
- *Конец* - время, до которого производится подсчет;
- *Период приведения* - коэффициент перевода в нужную единицу измерения времени. Например, если входная переменная - это расход, измеряемый в кубометрах в час, то при постоянном расходе в 1 куб. м в час и периоде интегрирования в 1 час резуль-

тат без учета периода приведения окажется 3600 куб. м вместо 1 куб. м. Это вызвано тем, что во всех вычислениях, связанных со временем, в MasterSCADA время считается в секундах. Поэтому для данного примера период приведения - это час, а значение периода равно 3600, так как оно должно быть задано в секундах.

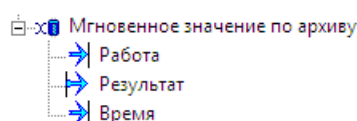
Работа ФБ описана формулой: $Результат = \frac{\sum_{i=1}^n (Значение_i * (Время_i - Время_{i-1}))}{Период\ приведения}$, где i - это последовательный номер, начиная с 1, всех значений в архиве от начального до конечного времени периода усреднения. Каждое значение в диапазоне умножается на разность времени записи данного значения и времени записи предыдущего значения. Все эти произведения суммируются и делятся на период приведения.

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Мгновенное значение по архиву

ФБ *Мгновенное значение по архиву* находит значение в архиве, удовлетворяющее заданному времени.

Вид функционального блока в дереве объекта:



Входы и выходы:

- *Работа* - признак включения ФБ в работу;
- *Результат* - запрашиваемое значение;
- *Время* - заданное время, по которому находится значения в архиве.

Параметры настройки:

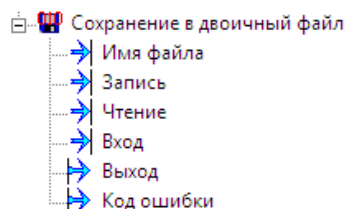
- *Имя параметра* - переменная, значение которой определяется данным ФБ;
- *Время* - время определения значения;
- *Тип поиска* - критерий поиска значения (*Ближайшее сверху*, *Ближайшее снизу*, *Ближайшее*).

ФБ выбирает значение по заданному в параметрах настройки критерию *Ближайшее по времени сверху*, *Ближайшее снизу*, *Ближайшее*. Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Сохранения в двоичный файл

По переднему фронту входов *Запись* сохраняет в файл значения параметра *Вход*. По переднему фронту входа чтение передает на *Выход* данные из ранее созданного файла.

Вид функционального блока в дереве объекта:



Файлы находятся в папке функционального блока (которая находится в папке объекта, в котором находится ФБ и имеет имя, совпадающее с именем ФБ). Таким образом, можно использовать для нескольких функциональных блоков одно и то же имя файла чтобы одновременно производить чтение нескольких наборов данных. С другой стороны, это ограничивает работу тем, что ФБ не может прочитать файл, созданный другим ФБ (если, не скопировать файл в нужную папку). Если файл невозможно открыть, формируется сообщение функционального блока и выход *Код ошибки* принимает соответствующее значение, которое сохраняется до следующей команды.

Код ошибки	Описание
0	Операция с файлом была успешной.
1	Пустая строка на входе <i>Имя файла</i> .
2	Ошибка при чтении файла.
3	Не бывает.
4	Ошибка при создании файла.

Если файл с указанным именем уже существует, то его перезапись происходит без предупреждения.

Входы и выходы:

	Имя	Тип	Примечание
Входы	Имя файла	Строковый	Имя файла для чтения или записи, без расширения (расширение .csv добавляется автоматически).
	Запись	Логический	Запись в файл начинается при переходе значения из <i>Ложь</i> в <i>Истина</i> (по переднему фронту).
	Чтение	Логический	Чтение из файла начинается при переходе значения из <i>Ложь</i> в <i>Истина</i> (по переднему фронту).
Выходы	Группа входов	Строковый	Данные для записи в файл.
	Группа выходов	Строковый	Данные, прочитанные из файла.
	Код ошибки	Целый	относится к последней проведенной операции с файлом.

Сообщения:

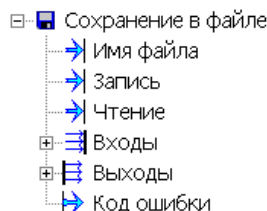
Сообщение	Условие
Неправильное имя файла	Пустая строка на входе <i>Имя файла</i> , при выполнении одной из команд.
Ошибка при открытии файла	Возникает если файл не существует, занят другим приложением, имеет атрибут <i>только чтение</i> и прочих ситуациях, когда запись или чтение невозможны.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Сохранение в файле

По переднему фронту входов *Запись* и (или) *Чтение* сохраняет в текстовый файл данные группы *Входы* и (или) передает на *Выходы* данные из ранее созданного текстового файла.

Вид функционального блока в дереве объекта:



Файлы находятся в папке функционального блока (которая находится в папке объекта, в котором находится ФБ и имеет имя, совпадающее с именем ФБ). Таким образом, возможно использовать для нескольких функциональных блоков одно и то же имя файла чтобы одновременно производить чтение нескольких наборов данных. С другой стороны, это ограничивает работу тем, что ФБ не может прочитать файл, созданный другим ФБ (если не скопировать файл в нужную папку).

- ❖ **Важно!** Входы и выходы имеют строковый тип. Таким образом, если используются числовые данные, позаботьтесь о преобразовании типа переменных, с которыми связываются выходы (тип данных, поступающих на вход преобразуется в строковый автоматически). Для этого если создаётся связь Выхода с переменной фиксированного типа, преобразование произойдет автоматически, в противном случае тип переменной, с которой создается связь выхода надо указать явно. Например, при перетаскивании выхода в формулу расчета, тип переменной, используемой в формуле сразу установится в строковый. Если необходимо чтобы переменная приняла числовой тип, необходимо изменить ее свойства!

В качестве разделителя данных при записи используется *Enter*, при чтении - *Enter* или *;*. Если количество данных в файле меньше, чем число выходов, то оставшимся Выходам присваиваются пустые строки (старые значения не сохраняются). Если файл невозможно от-

крыть, формируется сообщение функционального блока и выход Код ошибки принимает соответствующее значение, которое сохраняется до следующей команды.

Код ошибки	Описание
0	Операция с файлом была успешной.
1	Пустая строка на входе <i>Имя файла</i> .
2	Ошибка при чтении файла.
3	Не бывает.
4	Ошибка при создании файла.

Если файл с указанным именем уже существует, то его перезапись происходит без предупреждения. Свойство *Количество переменных* (целый тип) может принимать значение от 0 до 256.

Входы и выходы:

	Имя	Тип	Примечание
Входы	Имя файла	Строковый	Имя файла для чтения или записи, без расширения (расширение .csv добавляется автоматически).
	Запись	Логический	Запись в файл начинается при переходе значения из <i>Ложь</i> в <i>Истина</i> (по переднему фронту).
	Чтение	Логический	Чтение из файла начинается при переходе значения из <i>Ложь</i> в <i>Истина</i> (по переднему фронту).
Выходы	Группа входов	Строковый	Данные для записи в файл.
	Группа выходов	Строковый	Данные, прочитанные из файла.
	Код ошибки	Целый	относится к последней проведенной операции с файлом.

Сообщения:

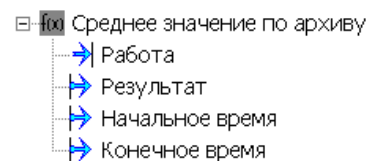
Сообщение	Условие
Неправильное имя файла	Пустая строка на входе <i>Имя файла</i> , при выполнении одной из команд.
Ошибка при открытии файла	Возникает если файл не существует, занят другим приложением, имеет атрибут <i>только чтение</i> и прочих ситуациях, когда запись или чтение невозможны.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Среднее значение по архиву

ФБ *Среднее значение по архиву* (соответствует функции *ArchAverage* в формуле) считает среднее значение по выбранной переменной (команде, значению, расчету) за заданный период времени. ФБ имеет один вход - *Работа* и три выхода - *Результат*, *Начальное время* и *Конечное время*.

Вид функционального блока в дереве объекта:



Входы и выходы:

- *Работа* - признак включения ФБ в работу;
- *Результат* - полученное среднее значение;
- *Начальное время* - фактическое время первого значения в архиве за заданный период времени;
- *Конечное время* - фактическое время последнего значения в архиве за заданный период времени.

Параметры настройки:

- *Имя параметра* - переменная, для которой вычисляется ФБ;
- *Начало* - время, начиная с которого производится подсчет;
- *Конец* - время, до которого производится подсчет.

Алгоритм работы ФБ: $Результат = \frac{\sum_{i=1}^n (Значение_i * (Время_i - Время_{i-1}))}{Конечное\ время - Начальное\ время}$, где i - это последовательный номер, начиная с 1, всех значений в

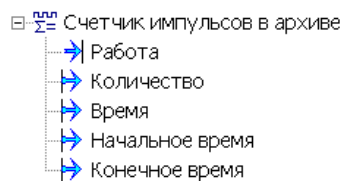
архиве от начального до конечного времени периода усреднения. Каждое значение в диапазоне умножается на разность времени записи данного значения и времени записи предыдущего значения. Все эти произведения суммируются и делятся на период времени усреднения.

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Счетчик импульсов и пробега в архиве

ФБ *Счетчик импульсов и пробега в архиве* (соответствует функции *ArchDiscreteCounter* в формуле) считает сколько раз выбранная переменная (команда, значение, расчет) за заданный период времени переходила в включенное состояние, а также какое суммарное время она находилась в этом состоянии (время пробега).

Вид функционального блока в дереве объекта:



Входы и выходы:

- *Работа* - признак включения ФБ в работу;
- *Количество* - счетчик импульсов;
- *Время* - время пробега (в сек.);
- *Начальное время* - фактическое время первого значения в архиве за заданный период времени;
- *Конечное время* - фактическое время последнего значения в архиве за заданный период времени.

Параметры настройки:

- *Имя параметра* - число переключений этой переменной вычисляется данным ФБ;
- *Начало* - время, начиная с которого производится подсчет;
- *Конец* - время, до которого производится подсчет.

Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Формирование отчета РАС

Функциональный блок используется для формирования отчетов *РАС* (регистрация аварийных событий). В отчет могут попадать значения параметров (защиты, состояние арматуры, МСН) на момент формирования отчета, либо все переходы значений параметров из одного состояния в другое за какой-либо отрезок времени. При перетаскивании ФБ в *Рапорт* (таблицу Excel), создается таблица. Вид таблицы зависит от способа формирования отчета.

Входы и выходы:

Название	Тип	Назначение
Входы		
Начало	Время	В случае формирования отчета <i>Изменение состояния</i> , задается время, с которого данные по параметрам начинают поступать из архива. MasterSCADA в отчет. В случае формирования отчета <i>Состояние на момент аварии</i> , определяет момент времени, за который нужно определить значение состояние параметра (защиты, состояние арматуры, МСН).
Конец	Время	В случае формирования отчета <i>Состояние на момент аварии</i> вход отсутствует. В случае формирования отчета <i>Изменение состояния</i> задается время, до которого данные по параметрам поступают в отчет.
Имя файла	Текстовый	Определяет место хранения и имя файла данных. По умолчанию, файл данных формируется во вложенной директории проекта, в персональной папке функционального блока.
Сформировать	Логический	По переднему фронту создается файл данных для отчета.
Группа входов: <i>Параметры</i>	Тип переменных в группе зависит от источника.	С данными входами устанавливается связь с переменными MasterSCADA, которые определяют состояния защит (тип перечислимый), арматуры, МСН (тип логический). Присоединенные переменные должны архивироваться.

Название	Тип	Назначение
Выходы		
Сформирован	Время	Носит информативный характер, сообщает время, когда ФБ сформировал файл данных.
Выполняется	Логический	Принимает значение <i>Истина</i> во время формирования отчета.
Ошибка	Логический	Если рапорт сформировать не удалось, то выход равен <i>Истина</i> .
Описание	Строковый	Выдается текст ошибки.

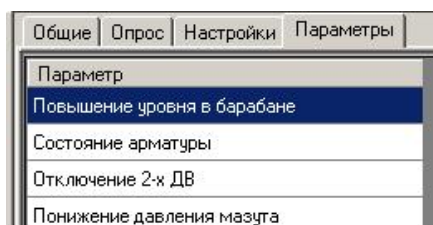
Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Страница свойств *Настройки* ФБ *Формирование отчета РАС*. *Количество параметров* определяет количество параметров (состояния защит, запорной арматуры, МСН), которые необходимы для составления отчета по каждой из защит.

Вид отчета определяет способ формирования отчета:

- *Изменение состояний*. Отчет представляет собой таблицу, в которой будут отражены все изменения параметров за отчетный период времени.
- *Состояние на момент аварии*. Для каждого параметра будет сформирована одна строка, в которой будет отображено состояние параметра в момент времени *Начало*.

Страница свойств *Параметры* изображена на рисунке ниже:



На странице свойств задаются имена параметров, с которыми они будут попадать в отчет, в столбец *Наименование*.

Формирование отчета РАС. Вид таблицы, которая будет формироваться, если выбран вид отчета *Состояние на момент аварии*.

Лист 1:

№	KKS	Наименование	Состояние
1	<KKS параметра 1>	<Наименование параметра>	<Состояние>
2	<KKS параметра 2>	<Наименование параметра>	<Состояние>
...
...	<KKS параметра n-1>	<Наименование параметра>	<Состояние>
n	<KKS параметра n>	<Наименование параметра>	<Состояние>

Вид таблицы, которая будет формироваться, если выбран вид отчета *Изменение состояния*. Лист 1:

№	KKS	Наименование	Изменение состояния	Время изменения
1	<KKS параметра 1>	<Наименование параметра>	<Состояние до> ® <Состояние после>	dd/mm/yy hh:mm:ss:ms
2	<KKS параметра 2>	<Наименование параметра>	<Состояние до> ® <Состояние после>	dd/mm/yy hh:mm:ss:ms
...
...	<KKS параметра n-1>	<Наименование параметра>	<Состояние до> ® <Состояние после>	dd/mm/yy hh:mm:ss:ms
n	<KKS параметра n>	<Наименование параметра>	<Состояние до> ® <Состояние после>	dd/mm/yy hh:mm:ss:ms

Формирование отчета РОП

ФБ служит для составления отчета РОП. При перетаскивании ФБ в *Рапорт* (таблицу *Excel*), создается таблица, строки которой соответствуют входным параметрам (количество параметров настраивается на странице свойств *Количество связей*). В режиме исполнения в этой таблице (по входу *Сформировать*) создаются дополнительные строки, в ячейки которых заносятся значения, рассчитанные в соответствии с алгоритмом обработки. Этот ФБ при составлении отчета использует ФБ *Обработка аналогового параметра*. Все данные для

обработки берутся с выходов этого ФБ. На вход ФБ *Обработка аналогового параметра* должен подаваться медленно изменяющийся сигнал (по отношению к периоду опроса), чтобы отчет РОП формировался корректно. Алгоритм выбора используемых ФБ - просматриваются все входы группы *Связи*, если вход имеет связь, то переменная, с которой установлена связь, попадает в отчет. ФБ *Обработка аналогового параметра* должен архивироваться иначе в протокол работы запишется ошибка.

Входы:

- *Начало* – время, определяющее, с какого момента необходимо обработать данные из архива;
- *Конец* – время, до которого необходимо обрабатывать данные по архиву;
- *Имя файла* – (текстовый) определяет под каким именем и где сохранять информацию, собранную данным ФБ (путь к файлу может быть как относительным, так и абсолютным). Если не указано, то по умолчанию информация сохраняется в файле *данные.txt* в папке проекта;
- *Сформировать* – (логический), по переднему фронту этого входа ФБ начинает собирать информацию и создает файл;
- *Группа Связи* - служит для того, чтобы установить связи с архивируемыми аналоговыми переменными, значения которых должны попасть в отчет.

Выходы:

- *Сформирован* – возвращает время создания файла для отчета;
- *Выполняется* – значение *Вкл.* означает, что ФБ обрабатывает данные;
- *Ошибка* – если имеет значение *Вкл.*, то при выполнении ФБ произошла ошибка;
- *Описание* – возвращает описание ошибки, если на выходе *Ошибка* значение *Вкл.* Оно также записывается в протокол работы.

У ФБ есть два режима работы: когда в отчет пишутся все отклонения параметра или когда создается обобщенный отчет. Режим работы определяет свойство *Тип ведомости*.

Таблица заполняется следующим образом:

- *Ведомость текущих отклонений параметров* (в каждой строке этой ведомости расчеты производятся для интервала времени [*Время нарушения; Время возврата*]):
 - *№* - порядковый номер события;
 - *Код параметра* - ККС ФБ *Обработка аналогового параметра*. Если ККС отсутствует, то название ФБ с путем к нему;
 - *Сокращенное наименование* - значение настройки *Сокращенное наименование ФБ Обработка аналогового параметра*;
 - *Размерность* - размерность шкалы выхода ФБ *Обработка аналогового параметра*;
 - *Мах отклонение* - разность между максимальным нарушенным значением и значением нарушенной границы;
 - *Среднее отклонение* - среднее отклонение параметра от границы;
 - *Нарушенная граница* - мнемоническое обозначение нарушенной регламентной границы (*НАГ, НПГ, ВПГ, ВАГ*);
 - *Граничное значение* - значение настройки ФБ *Обработка аналогового параметра* в зависимости от типа отклонения;
 - *Время нарушения* - метка нарушения регламентной границы (времени дд/мм/гг чч:мм:сс), фиксируется по факту перехода выхода *НАГ/ВАГ/НПГ/ВПГ* из *Ложь* в *Истина*;
 - *Время возврата* - метка времени в формате входа параметра в норму (времени дд/мм/гг чч:мм:сс), фиксируется по факту перехода выхода *НАГ/ВАГ/НПГ/ВПГ* из *Истина* в *Ложь*;
 - *Время максимального отклонения* - метка времени в формате значения параметра при максимальном отклонении от границы (времени дд/мм/гг чч:мм:сс);
 - *Продолжительность отклонения* - разность времени входа в норму и времени нарушения соответствующей границы;

Обобщенная ведомость отклонений параметров (в каждой строке этой ведомости расчеты производятся для всего интервала времени формирования отчета РОП, для каждого параметра возможно максимум 4 строки - *ВАГ, ВПГ, НПГ, НАГ*):

- *№* - порядковый номер события;
- *Код параметра* - ККС ФБ *Обработка аналогового параметра*. Если ККС отсутствует, то название ФБ с путем к нему;

- Сокращенное наименование - значение настройки *Сокращенное наименование* ФБ *Обработка аналогового параметра*;
- *Размерность* - размерность шкалы выхода ФБ *Обработка аналогового параметра*;
- *Мах отклонение* - разность между максимальным нарушенным значением и значением нарушенной границы;
- *Среднее отклонение* - среднее отклонение параметра от границы;
- *Нарушенная граница* - мнемоническое обозначение нарушенной регламентной границы (*НАГ, НПГ, ВПГ, ВАГ*);
- Граничное значение - значение настройки ФБ *Обработка аналогового параметра* в зависимости от типа отклонения;
- Продолжительность отклонения - суммарное значение всех продолжительностей отклонений данного типа, вычисляемых как разность времени входа в норму и времени нарушения соответствующей границы.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

Страница свойств *Настройки* ФБ *Формирование отчета РОП* выглядит следующим образом.



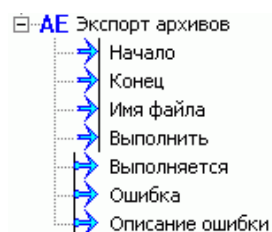
Свойство *Количество связей* определяет, по какому количеству переменных вести отчет. Свойство *Тип ведомости* имеет значения:

- Ведомость текущих отклонений параметров;
- Обобщенная ведомость отклонений параметров.

Экспорт архивов

ФБ *Экспорт архивов* предназначен для выполнения экспорта архива сообщений или данных за произвольный интервал в файл *.mdb*.

Вид функционального блока в дереве объекта:



Входы и выходы:

- *Начало* - нижняя граница интервала времени, за который надо экспортировать архив;
- *Конец* - верхняя граница интервала времени, за который надо экспортировать архив;
- *Имя файла* - наименование файла, в который производится экспорт. Файл должен иметь расширение *.mdb*. Может содержать относительный или абсолютный путь к нему. Если путь не указан, то файл формируется в папку *MasterSCADA Projects*;
- *Выполнить* - по переднему фронту производится экспорт;
- *Выполняется* - значение *Вкл.* на этом выходе означает, что идет процесс экспорта;
- *Ошибка* - значение *Вкл.* означает, что при экспорте произошла ошибка;
- *Описание ошибки* - возвращает описание ошибки, если на выходе *Ошибка* значение *Вкл.*;

Параметры настройки:

- *Вложенные объекты* - при значении *Истина* экспортируется не только архивные данные или сообщения объекта, которому принадлежит ФБ, но и всех вложенных в него объектов независимо от уровня вложенности;
- *Использовать архив компьютера* - при значении *Истина* осуществляется экспорт архива сообщений компьютера, на котором выполняется ФБ, или архива данных всех объектов, принадлежащих этому компьютеру;
- Тип архива - какой архив требуется экспортировать: архив данных или архив сообщений.

Поддерживаемый тип опроса - периодический. Восстановление при рестарте – не требуется.

Список строк


ВФБ *Список строк* предназначен для ввода и хранения пользовательских строковых данных в режиме исполнения. Как и любой ВФБ, *Список строк* состоит из функционального блока, располагающегося в дереве объектов и соответствующего визуального элемента, необходимого для управления этим ФБ из мнемосхемы. Данные, введенные в *Список строк* хранятся в текстовом файле, в папке объекта, которому принадлежит мнемосхема. Имя файла соответствует имени функционального блока в дереве. Этот файл можно редактировать во время режима исполнения, т.к. чтение данных из файла осуществляется каждый раз, при открытии выпадающего списка.

ФБ имеет два выхода. На выход *Выбранная строка* выводится выбранная пользователем строка, на выход *Номер выбранной строки* - её номер. На вход *Номер строки* можно подавать номер строки, при этом на выходе будет строка с запрошенным номером.

Элемент управления, представляющий *Список строк* в мнемосхеме (необходимо просто перетащить ФБ в мнемосхему) имеет вид:



Поле ввода предназначено для добавления новых строк, а также для отображения текущей выбранной строки. Выпадающий список содержит все ранее введенные строки.

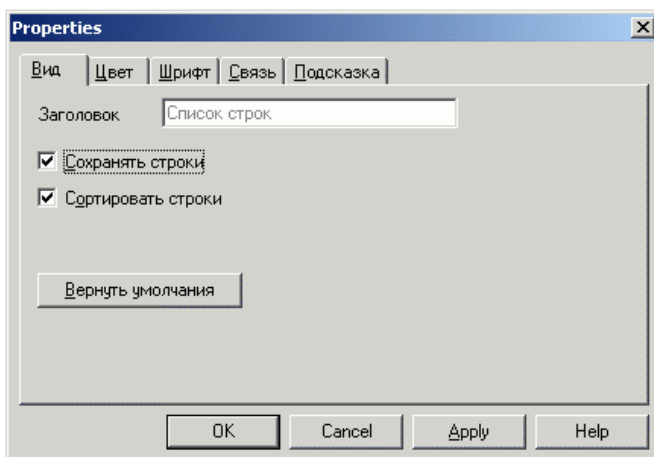
Кнопка  служит для следующих задач:

- Принятия ввода новой строки (в этом случае строка, введенная в поле ввода добавляется в список строк и автоматически появляется на выходе ФБ в дереве объектов);
- Подтверждения выбора существующей строки из выпадающего списка (на выходе ФБ строка появляется только по нажатии на эту кнопку).

Кнопка  предназначена для удаления текущей строки из списка строк.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – не требуется.

На закладке *Вид* настраиваются общие свойства *Списка строк*.



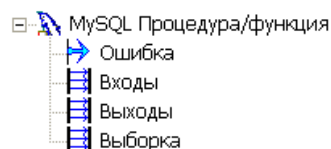
Поле ввода *Заголовок* - строка, которая будет отображаться в поле заголовка элемента. Флаг *Сохранять строки* устанавливает, сохранять введенные строки в файл, или нет. Если

флаг *Сортировать строки* установлен, то при открытии выпадающего списка, строки будут расположены в алфавитном порядке

My SQL процедура/функция

My SQL – система управления базами данных (СУБД) разработанная компанией Microsoft. Основным используемый язык запросов - Transact-SQL (расширенный вариант стандарта SQL-92).

Вид функционального блока в дереве объекта:



Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Страница свойств *Настройки* ФБ *My SQL Процедура/функция* выглядит следующим образом:

Элементы управления группы *Конфигурация*:

- *Имя процедуры* - задается имя процедуры;
- *Использовать выборку*:
 - *Считывать первые значения*;
 - *Считывать все значения*;
 - *Не использовать*;
- *Режим работы*:
 - *Запрос*;
 - *Процедура*;
- *Текст запроса* - задается текст запроса.

Группа подключение:

- *База* - имя созданной базы;
- *Пароль* - по умолчанию пароль отсутствует;
- *Пользователь* - по умолчанию пользователь *MySQL root*;
- *Сервер* - локальный *localhost* (в случае работы в сетевом проекте нужно указать IP-адрес).

Страница свойств *Параметры* ФБ *My SQL Процедура/функция* выглядит следующим образом:

Элемент управления *Процедура*:

- *Создать автоматически* - при нажатие кнопки из процедуры, созданной в базе данных, переносятся параметры в таблицу;
- *Название* - имя параметра;
- *Размер* - размер параметра;
- *Тип данных* - тип параметра;
- *Направление* - параметр может являться входным значением, выходным значением, или может быть входным и выходным значением одновременно.

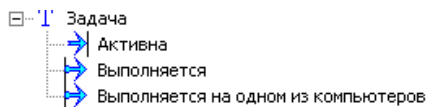
Элемент управления *Выборка*:

- *Название* - имя параметра;
- *Тип параметра* - тип параметра;
- *Метка времени* - выбранному параметру, присваивается метка времени, чтобы сделать выборку.

Служебные

Задача

Вид функционального блока в дереве объектов:



Функциональный блок предназначен для осуществления резервирования отдельных задач проектов, которые выполняются на нескольких независимых компьютерах и могут отличаться друг от друга.

Основная функциональность разбита на две части:

1. Механизм обмена статусов текущих задач между компьютерами;
2. Непосредственно ФБ *Задача*.

Вход *Активна* определяет, выполнять ли задачу на данном компьютере (значение этого входа принимается во внимание только при установленном флаге у *Компьютера* на странице свойств ФБ *Настройки*). При значении *Выкл.* задача на данном компьютере не будет выполняться.

Выход *Выполняется* выдает команду на выполнение задачи:

- *Вкл.* - выполнить задачу;
- *Выкл.* - не выполнять задачу.

Этот выход должен быть связан с соответствующим элементом задачи, выполнение которой требуется контролировать.

Механизм обмена статусов. Настройки производятся на странице свойств *Режим исполнения* у элемента *Компьютера* закладка *Связь - Резервирование компьютера*. Настройки системы вынесены в категорию *Резервирование задачи*, в ней настраивается количество компьютеров, на которых работает система и настройки для каждого компьютера: *ip-адрес* и *имя*. *Имя компьютера* - это уникальный идентификатор данного компьютера в системе. Для компьютера можно задать несколько ip-адресов (через запятую, пробел и точку с запятой).

При переходе в режим исполнения система определяет текущий компьютер, если при старте не получилось (например, у компьютера выключено сетевое подключение или неправильно настроен ip-адрес), то система простаивает до тех пор пока не сможет определить. После определения текущего компьютера система выполняет две функции: опрашивает все другие компьютеры о задачах, которые на них выполняются, и отвечает на аналогичные запросы других компьютеров.

Если при опросе компьютера произошла ошибка, то система пробует опросить его по другому ip-адресу (если он задан), когда ошибка опроса происходит по всем заданным адресам для компьютера система продолжает опрос с периодом в 5 минут, это сделано из-за того, что при ошибке опроса компьютера в некоторых конфигурациях сети поток опроса может

подвешиваться системой на 40 секунд, тем самым лишая другие компьютеры возможности быть опрошенными. Если другой компьютер опрашивает наш, то наш считает, что связь восстановлена и начинает опрашивать его с периодом в одну секунду.

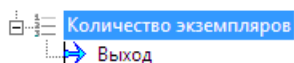
При старте ФБ регистрирует свою задачу в системе. ФБ предназначен для периодической работы или режима, когда на его вход постоянно подается значение. *Задача* может выполняться только на тех компьютерах, которые выделены флагом на закладке ФБ *Настройки* и на вход ФБ этих компьютеров *Активен* подано значение *Вкл.* Задача будет выполняться (выход ФБ примет значение *Вкл.*) на том из этих компьютеров, приоритет которого выше. Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Настройки ФБ сводятся к выбору, на каких компьютерах эта задача должна выполняться и приоритета компьютеров (чем выше в списке, тем важнее - задача будет выполняться на том компьютере, приоритет которого выше). Список компьютеров наследуется с закладки Компьютера *Режим исполнения/Связь*. Также у ФБ есть настройка *Задержка включения*, определяющая время, через которое выход ФБ сменит свое значение с *Выкл.* на *Вкл.* после срабатывания внутреннего алгоритма ФБ.

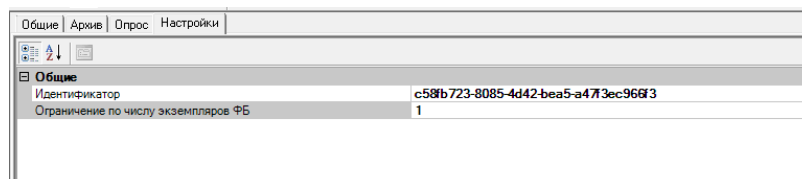
Количество экземпляров

Подсчитывает количество своих копий.

Вид функционального блока в дереве объекта:



Страница свойств *Настройки* ФБ *Количество экземпляров* выглядит следующим образом:



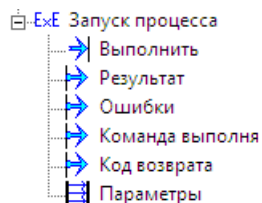
Поиск количества экземпляров данного ФБ осуществляется с помощью строки из поля *Идентификатор*. *Ограничение по числу экземпляров ФБ* - количество копий ФБ с указанным выше идентификатором.

Выход (логический) - если количество данных ФБ превышает настройку *Ограничение количество экземпляров ФБ* - *Ложь*, иначе *Истина*.

Запуск процесса

Этот ФБ позволяет запускать любые процессы непосредственно из окна MasterSCADA.

Вид функционального блока в дереве объекта:



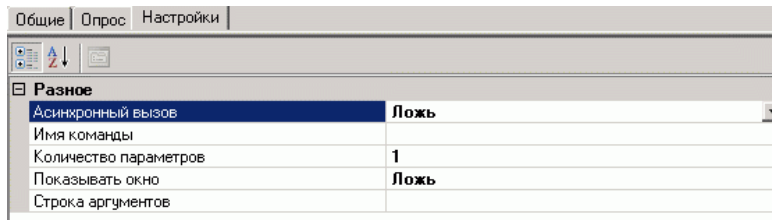
Входы и выходы:

	Обозначение	Тип	Описание
Входы	Выполнить	Логический	По переднему фронту осуществляется запуск процесса.
	Параметры	Изменяющийся тип	Параметры запуска процесса. Количество параметров определяется настройкой ФБ <i>Количество параметров</i> . Параметры используются в настройке <i>Строка аргументов</i> .
Выходы	Результат	Строковый	Выводится строка, содержащая информацию о результате выполнения команды.

Обозначение	Тип	Описание
Ошибки	Строковый	В случае ошибки выполнения команды выводится информация об ошибке.
Команда выполняется	Логический	Появляется, если в настройках задан асинхронный вызов. Отображает состояние выполнения команды.
Код возврата	Целый	Возвращает код ошибки.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Страница свойств *Настройки* ФБ *Запуск процесса* выглядит следующим образом:

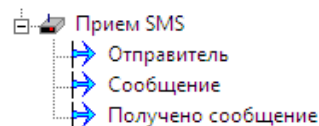


Если *Асинхронный вызов Вкл.*, то происходит вызов приложения осуществляется асинхронно, то есть выполнение программы MasterSCADA после вызова стороннего приложения не прерывается, если - *Выкл.*, то вызов приложения осуществляется синхронно, то есть после вызова стороннего приложения, цикл выполнения ФБ в MasterSCADA приостанавливается до тех пор, пока это приложение не будет выполнено. *Имя команды* - название исполняемого файла. Если это не стандартная программа Windows, то необходимо указывать еще путь к файлу. *Количество параметров* определяет количество входов в группе *Параметры*. *Показывать окно* - при значении *Вкл.* во время выполнения вызываемого приложения будет отображаться его окно, если это приложение консольное. Если требуется изменять параметры в режиме исполнения, есть возможность задавать их на входах группы *Параметры*. Тогда в строке аргументов в фигурных скобках записывается номер параметра в группе: например, {0}.

Прием SMS

Данный ФБ служит для получения SMS-сообщений.

Вид функционального блока в дереве объекта:



Данный ФБ работает с модемом на скорости 19200. Если в модеме отключено автоопределение скорости, то необходимо сменить её явно. Если поступление SMS не отображается на выходах ФБ, необходимо проверить настройку области памяти модема и убедиться что память не переполнена. Для этого необходимо воспользоваться программой *HyperTerminal*. Произведите подключение к модему и выполните команду: *AT+CPMS?*.

Модем должен прислать ответ вида: *+CPMS: «MT»,6,100,"ME",6,100,"SM",0,20*. Первым параметром должен идти *MT*, что означает что чтение SMS идет с памяти SIM карты и памяти модема. Второй и третий параметр - количество принятых и доступных SMS (в данном примере 6 из 100). Если первый параметр отличается (равен *SM* или *ME*), то необходимо настроить его на *MT* выполнив команду: *AT+CPMS="MT"*. Если память переполнена, то SMS сообщения нужно удалить. MasterSCADA автоматически удаляет из памяти модема сообщения полученные в процессе работы.

Входы ФБ:

Вход	Тип	Назначение
Отправитель	Строковый	Номер удаленного объекта.
Сообщение	Строковый	Текст сообщения.
Получено сообщение	Логический	<i>Истина/Ложь</i> - получено сообщение / не получено сооб-

щение.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Страница свойств *Настройки*:

Общие | Архив | Опрос | Настройки

Общие

Номер порта: 0

Номер порта - Номер *COM-порта* компьютера, к которому подключен модем.

Отправка e-mail

Данный функциональный блок служит для отправки электронных писем (e-mail).

Входы:

Обозначение	Тип	Назначение
Получатель	Строковый	Адрес получателя e-mail.
Тема	Строковый	Задается тема письма.
Текст	Строковый	Задается текст письма.
Послать письмо	Логический	Письмо отправиться по переднему фронту сигнала.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Страница свойств *Настройки* данного функционального блока выглядит следующим образом:

Общие | Опрос | Настройки

Аутентификация

Имя учетной записи

Пароль

Параметры

Отправитель

Сервер SMTP

Имя учетной записи

Применить | Отменить | Справка

Функции группы *Аутентификация*:

- *Имя учетной записи* - вводится имя учетной записи на почтовом сервере;
- *Пароль* - вводится пароль доступа для данной учетной записи.

Функции группы *Параметры*:

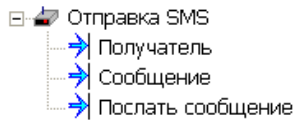
- *Отправитель* - задается имя отправителя электронного письма;
- *Сервер SMTP* - вводится имя почтового сервера для отправки e-mail.

Отправка *Email* поддерживает только расширение службы *SMTP* для *Secure SMTP* через протокол *TLS*, определенный в *RFC 3207*. В этом режиме сеанс *SMTP* начинается в незашифрованном канале, затем от клиента серверу дается команда *STARTTLS*, чтобы переключиться на безопасное соединение с помощью протокола *SSL*. Метод альтернативного подключения - метод, при котором сеанс *SSL* устанавливается заранее, до отправки команд протокола. Этот способ подключения иногда называют *SMTP/SSL*, *SMTP* через *SSL* или *SMTPS*. По умолчанию для него используется *порт 465*.

Отправка SMS

Данный функциональный блок служит для отправки SMS сообщений. Данный ФБ работает с модемом на скорости 19200. Если в модеме отключено автоопределение скорости, то необходимо сменить её явно.

Вид функционального блока в дереве объекта:

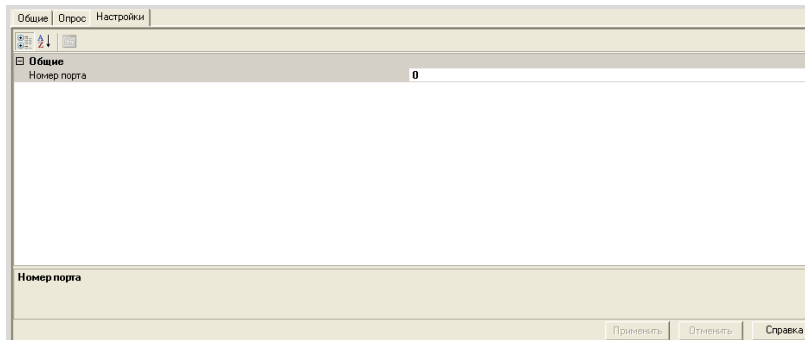


Входы:

Обозначение	Тип	Назначение
Получатель	Строковый	Номер получателя.
Сообщение	Строковый	Задается текст сообщения.
Послать сообщение	Логический	Сообщение отправиться по переднему фронту сигнала.

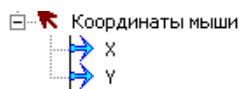
Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Страница свойств *Настройки* ФБ *Отправка SMS* выглядит следующим образом:



Координаты мыши

Позволяет в режиме исполнения определить координаты курсора. Вид в дереве объектов:



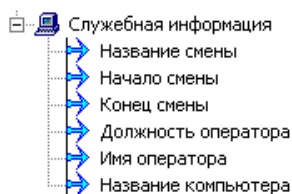
Входы:

- X (целый) - значение положения курсора по оси X;
- Y (целый) - значение положения курсора по оси Y.

Результат значений определяется в пикселях. Отсчет значений начинается с верхней точки экрана.

Служебная информация

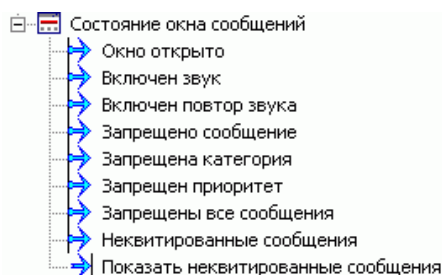
На *Выходах* функционального блока выводится соответствующая служебная информация. Вид функционального блока в дереве объекта:



Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется.

Состояние окна сообщений

ФБ *Состояние окна сообщений* служит для отображения настроек окна сообщений. Поддерживаемый тип опроса - периодический. Восстановление при рестарте - не требуется. Вид функционального блока в дереве объекта:



Входы/Выходы:

- *Окно открыто* - имеет значение *Вкл.* при открытом окне сообщений;
- *Включен звук* - имеет значение *Вкл.* при включенной настройке *Звук* в окне сообщений;
- *Включен повтор звука* - имеет значение *Вкл.* при включенной настройке *Повтор* в окне сообщений.

Следующие выходы служат для индикации соответствующих запретов сообщений:

- Запрещено сообщение;
- Запрещена категория;
- Запрещен приоритет;
- Запрещены все сообщения.

Следующие переменные появляются у ФБ, если настройка ФБ *Отображать наличие неквитированных сообщений* имеет значение *Истина*:

- *Неквитированные сообщения* - имеет значение *Вкл.*, если в проекте имеются неквитированные сообщения;
- *Показать неквитированные сообщения* - по переднему фронту открывается *Основной журнал Компьютера*, в котором показаны эти сообщения.

Управление документом

Позволяет производить в режиме исполнения такие действия над определенным документом, как: открытие, сохранение, печать, с возможностью задания различных параметров этих действий, а также отображает состояние документа.

Входы:

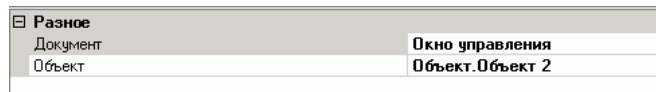
Обозначение		Тип	Описание
Группа <i>Открыть</i>	Открыть	Логический	По переднему фронту открывается документ. Если сработал передний фронт на входе <i>Открыть</i> , то документ открывается при значении <i>Вкл.</i> – как окно, <i>Выкл.</i> – как документ.
	Как окно	Логический	
	Левый	Целый	
	Верхний	Целый	
	Ширина	Целый	
Группа <i>Сохранить</i>	Высота	Целый	Если документ открывается как окно, то высота окна.
	Сохранить	Логический	По переднему фронту сохраняет документ в файле на компьютере аналогично тому, как это делается при нажатии кнопки <i>Сохранить</i> на странице свойств соответствующего документа.
	Имя файла	Строковый	Если указано, то при нажатии кнопки <i>Сохранить</i> документ будет сохраняться в этот файл.
Группа <i>Распечатать</i>	Параметр	Строковый	Параметр сохранения документа. Например, масштаб тренда.
	Распечатать	Логический	По переднему фронту производится печать документа на принтере.
	Параметр	Строковый	Не используется.
	Закрыть	Логический	По переднему фронту документ закрывается.

Выходы:

Обозначение	Тип	Описание
Выполняется	Логический	Если над документом в настоящий момент выполняется какое-либо из действий при помощи ФБ, то имеет значение <i>Вкл.</i>
Ошибка	Строковый	В случае ошибки возвращает ее описание.
Открыт	Логический	Если документ открыт, то имеет значение <i>Вкл.</i>
Открыт как окно	Логический	Если документ открыт как окно, то имеет значение <i>Вкл.</i>
Левый	Целый	Если документ открыт как окно, то возвращает координату левого верхнего угла на оси абсцисс X.
Верхний	Целый	Если документ открыт как окно, то возвращает координату левого верхнего угла на оси ординат Y.
Ширина	Целый	Если документ открыт как окно, то возвращает значение ширины окна, в пикселях.
Высота	Целый	Если документ открыт как окно, то возвращает значение высоты окна, в пикселях.
Номер монитора	Целый	Если документ открыт как окно, то возвращает номер монитора, на котором окно открыто.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Страница свойств *Настройки ФБ Управление документом* выглядит следующим образом:



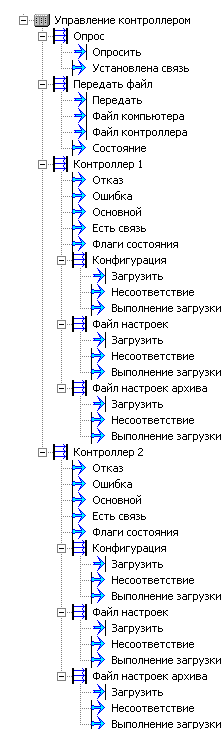
В поле *Документ* из выпадающего списка выбирается любой документ объекта, указанного в поле *Объект*. Именно над этим документом в режиме исполнения производятся различные действия при помощи данного функционального блока.

В поле *Объект* указывается название объекта, которому принадлежит документ. По умолчанию это объект, в котором находится ФБ. Изменить название можно путем перетаскивания нужного объекта из дерева в это поле.

Управление контроллером

Функциональный блок предоставляет информацию о состоянии резервированной пары контроллеров, включенных в проект. На выходах функционального блока отображается состояние диагностики контроллера. ФБ возможно использовать только при наличии контроллера с предустановленной системой MasterPLC.

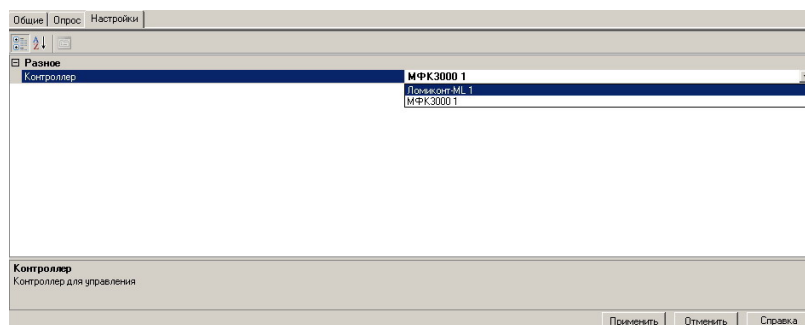
Вид в дереве:



Описание входов и выходов:

Название	Тип	Назначение
Группа Опрос		
Используется в случае, если связь с контроллерами установлена через GSM-модем.		
Группы Контроллер 1 и Контроллер 2		
Отказ	Выход	Логический
Выход равен единице, когда один или более модулей диагностики имеет статус <i>FAIL</i>		
Ошибка	Выход	Логический
Выход равен единице, если один или более модулей диагностики имеет статус <i>ERR</i>		
Основной	Выход	Логический
Выход равен единице, если контроллер работает в режиме <i>Основной</i> . Выход равен нулю, если контроллер работает в режиме резервный.		
Есть связь	Выход	Логический
Выход равен единице, если связь с контроллером установлена. При потере связи выход равен нулю.		
Флаги состояния	Выход	Беззнаковый целый
Группа Конфигурация		
Загрузить	Вход	Логический
Если выход равен единице, то выполняется загрузка конфигурации контроллера		
Несоответствие	Выход	Логический
Выход равен единице, если конфигурация контроллера не соответствует конфигурации компьютера.		
Выполнение загрузки	Выход	Логический
Выход равен единице, если выполняется загрузка конфигурации.		
Группа Файл настроек		
Загрузить	Вход	Логический
Если выход равен единице, то выполняется загрузка файла настроек контроллера.		
Несоответствие	Выход	Логический
Выход равен единице, если файл настроек контроллера не соответствует файлу настроек компьютера.		
Выполнение загрузки	Выход	Логический
Выход равен единице, если выполняется загрузка файла настроек.		
Группа Файл настроек архива		
Загрузить	Вход	Логический
Если выход равен единице, то выполняется загрузка файла настроек архива контроллера.		
Несоответствие	Выход	Логический
Выход равен единице, если файл настроек архива контроллера не соответствуют файлу настроек архива компьютера.		
Выполнение загрузки	Выход	Логический
Выход равен единице, если выполняется загрузка файла настроек архива.		
Группа Исполнительная система		
Загрузить	Вход	Логический
Если выход равен единице, то выполняется загрузка исполнительной системы контроллера.		
Несоответствие	Выход	Логический
Выход равен единице, если исполнительная система контроллера не соответствует исполнительной системе, хранящейся на компьютере.		
Выполнение загрузки	Выход	Логический
Выход равен единице, если выполняется загрузка исполнительной системы.		
Группа Контроль перезагрузки		
Перезагрузка	Выход	Логический
Если выход равен единице, то выполняется перезагрузка контроллера.		
Причина перезагрузки	Выход	Целый
Описывает код перезагрузки. Содержит цифровой код причины. 0 - причина неизвестна. 1 - причиной является загрузка конфигурации.		
Описание перезагрузки	Выход	Строковый
Строковое описание причины.		

Страница свойств Настройки ФБ Управление контроллером



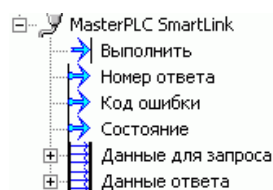
В поле *Контроллер* выбирается, тот контроллер, который находится в проекте и информацию о состоянии которого необходимо получать через данный функциональный блок. Отдельные выходы функционального блока можно размещать на мнемосхеме в виде индикаторов.

Master PLC SmartLink

MasterPLC Smartlink предназначен для обмена данными через COM-порт с произвольным устройством. Универсальность достигается тем, что протокол обмена описывается в настройках ФБ с помощью специальных шаблонов, в которых можно задавать последовательность посылаемых и ожидаемых байт данных, подставляя в эти пакеты значения входов ФБ и распределять полученные в ответе данные по выходам. Блок может работать как с текстовыми, так и с бинарными протоколами. Благодаря достаточно гибким шаблонам есть возможность практически произвольно задавать представление данных в пакете – длину, порядок битов, тип данных. На данный момент ограничением является то, что пакеты должны быть фиксированной длины.

Нередко устройства на один запрос могут отвечать различными по структуре пакетами, например, отправка данных часто отличается от отправки сообщения об ошибке. Поэтому в ФБ на описание одного шаблона запроса к устройству, можно создать несколько возможных шаблонов-ответов. Если шаблон запроса не задан, ФБ работает в режиме прослушивания. Также можно реализовать работу по отправке пакетов без ожидания ответа. На основе шаблона запроса ФБ формирует сообщение (кадр), которое посылается в канал связи с устройством (COM-порт). Получив ответ на запрос, ФБ разбирает его, сопоставляет результат, выделяет данные, которые нужно записать на выходы ФБ, возможна проверка контрольной суммы. Если ответ подходит к одному из шаблонов, то производится запись выходов.

Вид функционального блока в дереве объекта:



Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

Входы и выходы:

	Имя	Тип	Примечание
Входы	Выполнить	Логический	По переднему фронту производится чтение данных из группы входов, и отправка запроса в канал связи (COM-порт).
	Входные данные	(группа с переменным числом входов)	С входов этой группы читаются данные для составления запроса. До отправки запроса все входы этой группы, должны получить значения, иначе запрос не будет составлен.
Выходы	Номер ответа	Целый	Можно задать несколько шаблонов ответов, при этом переменной <i>Номер ответа</i> будет присвоен номер шаблона, который подошел при разборе ответа.
	Код ошибки	Целый	0 – все в порядке; 1 – несовпадение контрольной суммы; 2 – длина ответа меньше ожидаемой; 3 – байты данных не соответствуют ожидаемым (в статическом выражении); 4 – внутренняя ошибка (не должна возникать, если возникла,

Имя	Тип	Примечание
		нужно обратиться к разработчику); 5 – ошибка синтаксиса в строке шаблона; 6 – превышено время ожидания; 7 – обращение к несуществующему входу/выходу (например, в шаблоне указано прочитать данные с входов с нулевого по седьмой, а входов всего три); 8 – строка не преобразовалась в число; 9-не правильные значения меток, может возникнуть например, если в выражении-функции переставлены местами первая и вторая метки;
Состояние	Целый	0 – нет запроса, ФБ не выполняет никаких операций; 1 – ожидание ответа.
Выходные данные	(группа с переменным числом выходов)	На выходы группы записываются значения, выделенные из полученного ответа от устройства. Каким образом выделять данные из пакета и в каком виде их представлять, описывается в шаблоне ответа.

Содержание кадров запросов или ответов можно условно разбить на части, как это обычно делается в описаниях протоколов (например: адрес, сумма байт данных, значения, контрольная сумма, и. т. п.) Каждая часть кадра соответствует выражению в шаблоне. Можно выделить следующие части, из которых состоят кадры и которые необходимо описывать:

- Статический (символ начала, конца, адрес и т.п.);
- Данные входов-выходов ФБ;
- Результат вычислений, зависящий от текущего содержания кадра (CRC, количество байт).

Для описания шаблона кадра используется строка специального формата. Далее перечислены выражения, с помощью которых описываются шаблоны.

Статические выражения. Используются для описания тех частей кадра, которые не меняются в режиме исполнения, например: символ начала кадра, адрес опрашиваемого устройства. Статическое выражение помещается в кадр (а при получении ответа - сравнивается) в виде ASCII кодов символов, записанных пользователем в шаблоне. Это наиболее удобно для текстовых протоколов. Если же требуется употребить в выражении бинарные данные, используйте знак '1' перед значением каждого байта, записанного в шестнадцатеричном формате ($\backslash xx$ – символ с кодом xx).

Можно также использовать следующие символы:

- $\backslash r$ – $\langle CR \rangle$ - возврат каретки (код 0D);
- $\backslash n$ - $\langle LF \rangle$ - перевод строки;
- $\backslash \backslash$ - символ \backslash .

Пример:

Шаблон	Кадр (пробелы вставлены для наглядности)
123\r	31 32 33 0D
\31\32\33\0D	31 32 33 0D

Специальные выражения. Специальные – это все выражения кроме статических. Они должны быть обрамлены фигурными скобками - $\{ \text{выражение} \}$. Не допускается рекурсия (т.е. выражения, вложенные, в другие выражения).

Метка. Выражения - метки не попадают в кадр запроса и не ожидаются в кадрах ответов. Они используются для выделения какого-то участка кадра. Например, контрольная сумма часто рассчитывается без учета байт самой контрольной суммы, а также начального и конечного символов. Чтобы пометить определенные места, в шаблоне употребляются метки с идентификаторами, а эти идентификаторы затем используются в тех выражениях шаблона, где необходимо указать отмеченный метками участок.

Общий вид $\{L:\langle ID \rangle\}$, где:

- L – символ, указывающий, что выражение является меткой, через двоеточие указывается;
- $\langle id \rangle$ - идентификатор метки. Идентификатором метки может быть любой символ от 0 до 9 (то есть максимально 10 меток).

➤ **Например:** $xxxx\{L:0\}xxxx\{L:5\}xxxx\{L:2\}$.

Значения входов ФБ в запросах и выходов ФБ в ответах. Ввиду многообразия существующих протоколов, для значений необходимо указывать несколько атрибутов, позволяющих его правильно интерпретировать в(из) последовательность байт.

Общий вид: $\{P(\langle \text{Нач_вх} \rangle - \langle \text{Кон_вх} \rangle) : \langle \text{Длина_в_кадре} \rangle \langle \text{Формат_кадра} \rangle : \langle \text{Формат_данных} \rangle\}$, где:

- $\langle \text{Нач_вх} \rangle$ и $\langle \text{Кон_вх} \rangle$ - это индексы начального и конечного входов/выходов ФБ в группе, индексирование начинается с 0;
- $\langle \text{Длина_в_кадре} \rangle$ – количество байт, которое занимает в кадре одно передаваемое значение;
- $\langle \text{Формат_кадра} \rangle$ - как представлены данные в кадре. Может принимать одно из 2-х значений:
 - b – бинарный. В кадре находятся сами байты, из которых состоит значение. При использовании бинарных протоколов обратите внимание на свойство *High-Low*. Если свойство *High-Low* установлено истина, то число помещается в кадр старшим байтом вперед, если ложь, то младшим байтом вперед;
 - t – текст. В кадре содержатся коды цифр значения. Например, для передачи числа 5 в кадр помещается число пятьдесят три (в шестнадцатеричном представлении, младшим байтом вперед 0x35) – значение символа '5' в кодировке ASCII. Перед этим может быть вставлено необходимое количество нулей, если $\langle \text{длина в кадре} \rangle$ превышает количество символов числа.
- $\langle \text{Формат_данных} \rangle$ - это поле зависит от того, какой указан формат кадра.

$\langle \text{Формат_данных} \rangle$ может принимать значения:

- Для бинарных форматов кадров (b):
 - i – данные представляют целое число;
 - f – данные представляют 4-байтное вещественное число. Длина в кадре в этом случае должна быть равна 4-м;
- Для текстовых (t):
 - d – означает, что символы в кадре представляют собой число в десятичной системе;
 - h - символы в кадре представляют собой число в шестнадцатеричной системе, цифры $a-f$ в нижнем регистре;
 - H - символы в кадре представляют собой число в шестнадцатеричной системе, цифры $A-F$ в верхнем регистре;
 - i – при разборе выражения типа $\{P(_ _):_t:i\}$ начальные символы, не являющиеся цифрами, не будут игнорироваться, кроме пробелов и нулей, а также распознается знак;
 - f – символы в кадре представляют десятичное вещественное число с точкой, отделяющей дробную часть. Вещественные числа, содержащие в тексте десятичную точку (запятую), также распознаются со знаком, несмотря на пробелы « $_$ » в начале или в конце, например: « -0.123 »; « $+123.456$ »; « $_123.456$ ».
- **Например:** текстовый кадр содержит байты 31 32, что соответствует символам 1 и 2. Если указано, что формат данных десятичный (d), то значение будет распознаваться как двенадцать, если шестнадцатеричный (h), то значение – семнадцать.

Формула (Контрольная сумма, Сумма байт) $\{F\langle \text{Идентификатор_формулы} \rangle (\langle \text{ид_н_м} \rangle - \langle \text{ид_к_м} \rangle) : \langle \text{Длина_в_кадре} \rangle \langle \text{Формат_кадра} \rangle : \langle \text{Формат_данных} \rangle\}$, где $F\langle \text{Идентификатор_формулы} \rangle$ определяет по какому из заданных алгоритмов вычисляется значение. Может принимать значения:

- $Fcrc1$ – сумма байт без переноса (остаток от деления суммы байт на 256);
- $Fcrc2$ – контрольная сумма *CRC16*, применяемая в протоколах, подобных *Modbus RTU*;
- $Flen$ – просто количество байт;
- $Fxor$ – исключаящее или всех байт между метками.

$\langle \text{ид_н_м} \rangle - \langle \text{ид_к_м} \rangle$ - идентификаторы меток. Значение формулы вычисляется на основе байт, лежащих между первой и второй меткой.

- ❖ **Важно!** Требуется чтобы в строке шаблона оба выражения-метки, используемые в выражении-формуле находились слева от выражения-формулы.

Остальные поля описывают представление значения в массиве байт и полностью идентичны выражению *Значения входов*.

Например:

В шаблоне

В кадре

{L:3}\01\02\03{L:5}{Fcrc1(3-5):1b:i}	01 02 03 06
{L:3}\01\02\03{L:5}{Fcrc1(3-5):4t:h}	01 02 03 30 30 30 36
{L:1}\00\00\00{L:2}{Flen(1-2):4t:d}	00 00 00 30 30 30 33

Страница свойств *Настройки* ФБ *MasterPLC SmartLink* выглядит следующим образом:

Описание свойств представлено в таблице ниже.

Свойство	Описание
Макс. число команд	Если от устройства не получен ответ, или к ответу не удастся применить ни одного шаблона, то запрос посылается повторно. Свойство указывает, сколько раз повторять запрос при ошибочных или отсутствующих ответах. Если шаблон ответа не задан, запрос будет отправляться один раз, независимо от значения этого свойства.
Приоритет запроса	Может принимать значения 1 или 2. Как правило, приоритет 1 указывается для запросов, посвященных чтению каких-то данных; приоритет 2 указывается, если запрос посылает команду в прибор. Запросы с приоритетом 2 выполняются раньше запросов с приоритетом 1.
Число входов	Количество входов, значения которых нужно помещать в запрос. К моменту создания запроса все эти входы должны получить значения. При подсчете связей, наличие входа будет считаться за связь, вне зависимости от того установлена связь или нет.
Число выходов	Количество выходов, на которые записываются данные, извлеченные из ответной посылки. При подсчете связей, наличие выхода будет считаться за связь, вне зависимости от того установлена связь или нет.
HighLow	Указывает, каким образом в запрос помещаются числа в бинарном виде. Если значение свойства <i>Истина</i> , то первым байтом вперед, <i>Ложь</i> – последним байтом вперед.
Время ожидания	Время, в течение которого ожидается ответ от устройства.
Максимальная пауза	если пакет не приходит сразу весь, а частями, в этом свойстве указывается максимальное время ожидания продолжения пакета.
Номер СОМ-порта	Номер СОМ-порта.
Окончание кадра	Если протокол обмена предусматривает специальный символ как признак окончания пакета, его следует задавать в этом свойстве. В шаблонах этого символа быть не должно, так как он будет добавляться/отрезаться независимо от составления/разбора пакета по шаблону. В общем случае конец пакета определяется как истечение времени ожидания ИЛИ появление символа окончания кадра.
Шаблон запроса	Строка специального формата, по которой составляется пакет запроса.
Шаблоны ответов	Строка специального формата, по которой разбирается пакет ответа. Если на запрос возможны несколько вариантов ответа, следует записать их все, разделяя шаблоны выражением <i>{Next}</i> . При написании слова <i>Next</i> обязательно учитывать регистр букв.

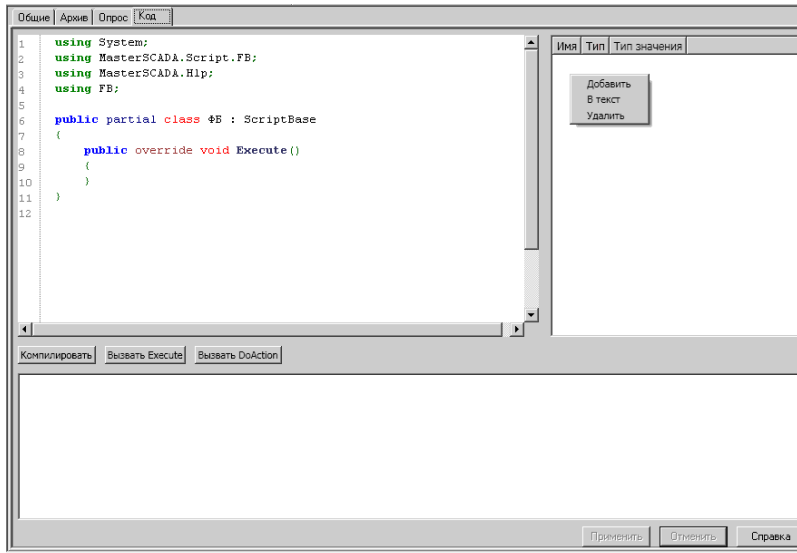
Скрипт

Функциональный блок *Скрипт* предназначен для передачи данных, позволяет записывать алгоритм выполнения на языке *C#*. Разработчик при использовании данного функционального блока должен иметь опыт программирования на языке *C#*.

Вид функционального блока в дереве объекта:



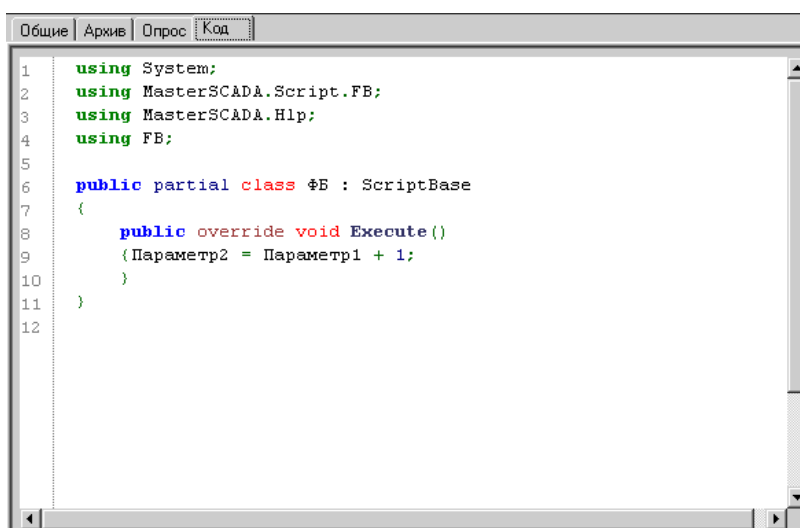
Входы и выходы ФБ *Скрипт* формируются на закладке *Код* страницы свойств. Из контекстного меню можно добавлять/удалять параметры. Также можно помещать имя параметра в текст программы (из контекстного меню или по щелчку).



В правом верхнем поле окна представлен *Список параметров*, который формирует входы и выходы ФБ *Скрипт*. Здесь назначается *Имя Параметра*, определяется его тип (*Запись*, *Чтение*, *Чтение/Запись*), выбирается *Тип его значения* (*Вещественный*, *Логический*, *Строковый* и т. д.).

Имя	Тип	Тип значения
Параметр1	Запись	Вещественный
Параметр2	Чтение	Целый
Параметр3	ЧтениеЗапись	Вещественный

В левом верхнем поле представлен *Редактор* для ввода текста программы. Алгоритм выполнения записывается в теле процедуры *Execute()*. Например так:

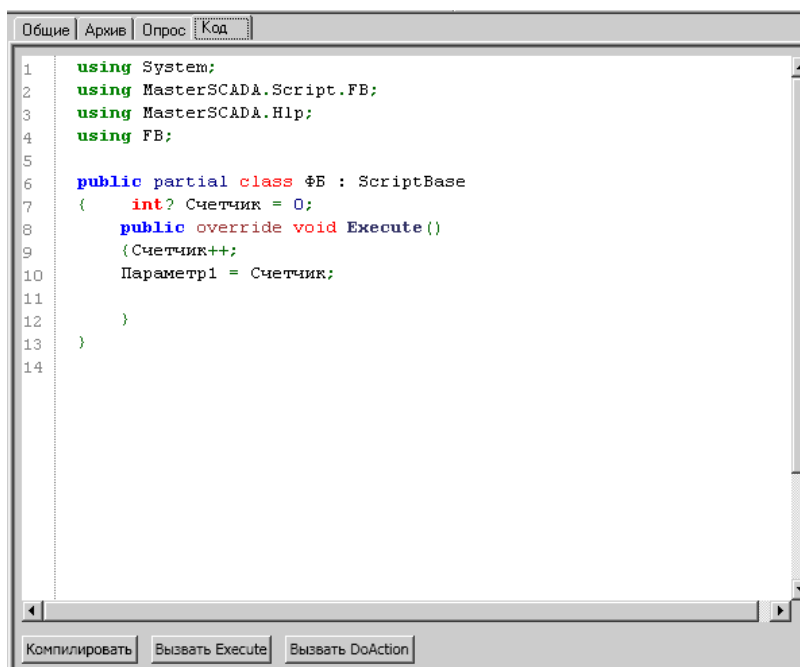


```

1  using System;
2  using MasterSCADA.Script.FB;
3  using MasterSCADA.Hlp;
4  using FB;
5
6  public partial class ФБ : ScriptBase
7  {
8      public override void Execute()
9      {
10         Параметр2 = Параметр1 + 1;
11     }
12 }

```

В теле класса можно объявлять переменные, чтобы их использовать в процессе работы. Они будут сохраняться между вызовами *Execute*:



```

1  using System;
2  using MasterSCADA.Script.FB;
3  using MasterSCADA.Hlp;
4  using FB;
5
6  public partial class ФБ : ScriptBase
7  {
8      int? Счетчик = 0;
9      public override void Execute()
10     {
11         Счетчик++;
12         Параметр1 = Счетчик;
13     }
14 }

```

Компилировать Вызвать Execute Вызвать DoAction

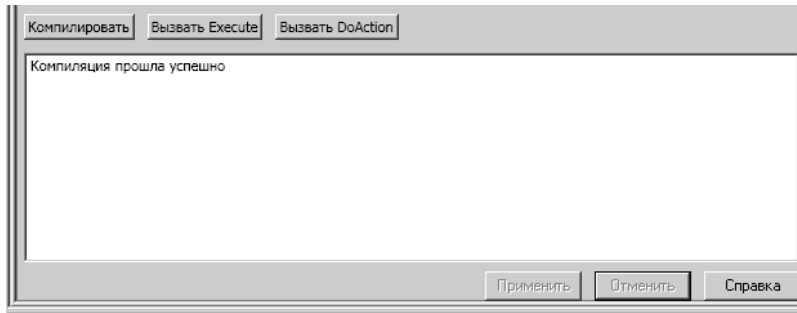
Если необходимо вызвать код при переходе в *Runtime*, нужно переопределить метод *Start*:

```

public override void Start()
{
    //здесь размещаете код
}

```

Аналогично нужно переопределить метод *Stop* для выполнения при выходе из *Runtime*. При объявлении переменных нужно дописывать к типу данных *?* (например *int?*). Это означает, что значение переменной может быть неопределено. Все параметры имеют такие типы (*int?*, *double?*), поэтому при записи арифметических операций с параметрами нужно учитывать, что тип этой операции тоже может быть неопределен. В нижнем поле отображается результат компиляции программы:

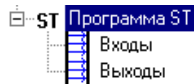


Кнопка *Компилировать* - компилирует код. Ошибки выдаются в нижнем поле. Кнопка *Вызвать Execute* - компилирует код и один раз вызывает *Execute*. Ошибки компиляции или выполнения выдаются в нижнем поле. Кнопка *Вызвать DoAction* - компилирует код и один раз вызывает *DoAction*. Этот метод может вызываться только по этой кнопке, в режиме исполнения он не используется. Он спроектирован для автоматизации разработки проекта в среде MasterSCADA (в нем можно выполнять различные операции с проектом). Для автоматизации разработки проектов предоставляется возможность обращения к объектной модели MasterSCADA в методе *DoAction*. Обращение к объектной модели MasterSCADA может нарушать безопасность выполнения MasterSCADA, поэтому налагает на разработчиков ответственность за надежность кода. Для доступа к проекту можно использовать свойства *HostFB.TreelItemHlp.Project*.

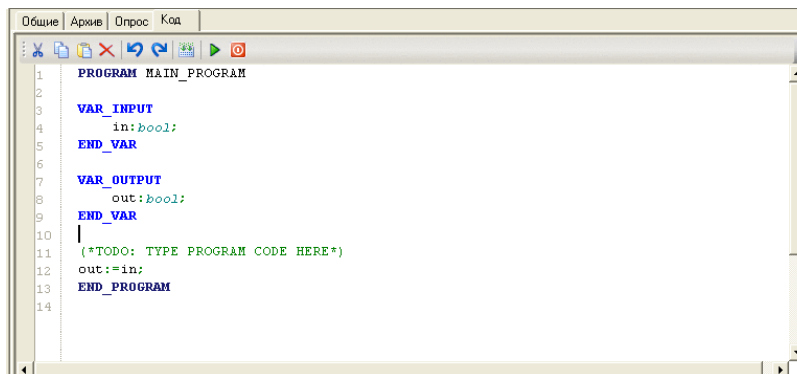
Программа ST

Данный функциональный блок позволяет разрабатывать собственные функциональные блоки и программы на языке ST стандарта МЭК-61131-3. Функциональный блок может исполняться как в компьютере, так и в программировании контроллера с исполнительной системой MasterPLC (применимость для конкретных контроллеров нужно уточнять в службе технической поддержки). Один из вариантов использования данного ФБ описан ниже.

После добавления *Программы ST* в дерево объектов как показано на рисунке ниже



пользователю во вкладке *Код* становится доступно поле для редактирования кода ST.



В данном поле пользователь имеет возможность создавать пользовательские типы, глобальные переменные, функции, функциональные блоки и программы на языке ST для использования в данном функциональном блоке.

```

Общие | Архив | Опрос | Код
1 FUNCTION_BLOCK DERIVATIVE
2 VAR_INPUT
3   RUN : BOOL ; (* 0 = reset *)
4   XIN : REAL ; (* Input to be differentiated *)
5   CYCLE : TIME ; (* Sampling period *)
6 END_VAR
7 VAR_OUTPUT
8   XOUT : REAL ; (* Differentiated output *)
9 END_VAR
10 VAR X1, X2, X3 : REAL ; END_VAR
11 IF RUN THEN
12   XOUT := (3.0 * (XIN - X3) + X1 - X2)
13         / (10.0 * TIME_TO_REAL(CYCLE)) ;
14   X3 := X2 ; X2 := X1 ; X1 := XIN ;
15 ELSE XOUT := 0.0 ; X1 := XIN ; X2 := XIN ; X3 := XIN ;
16 END_IF ;
17 END_FUNCTION_BLOCK
18
19

```

Для совместного использования описанных ранее типов, функциональных элементов пользователю рекомендуется использовать механизм библиотек, библиотеки подключаются к проекту на закладке *Библиотек - Библиотеки ST* у элемента *Система*. После описания функциональных элементов пользователю следует определить, с какого функционального элемента начнется выполнение работы. От стартового элемента требуется, чтобы типы входных и выходных элементов соответствовали типам используемых в MasterSCADA (вещественные, целочисленные, логические, время). Стартовым функциональным элементом считается либо программа с именем *MAIN_PROGRAMM*, либо первая определенная программа, либо первый определенный функциональный блок.

```

Общие | Архив | Опрос | Код
1 FUNCTION_BLOCK Main_program
2 VAR_INPUT
3   RUN : BOOL ; (* 0 = reset *)
4   XIN : REAL ; (* Input to be differentiated *)
5   CYCLE : TIME ; (* Sampling period *)
6 END_VAR
7 VAR_OUTPUT
8   XOUT : REAL ; (* Differentiated output *)
9 END_VAR
10 VAR X1, X2, X3 : REAL ; END_VAR
11 IF RUN THEN
12   XOUT := (3.0 * (XIN - X3) + X1 - X2)
13         / (10.0 * TIME_TO_REAL(CYCLE)) ;
14   X3 := X2 ; X2 := X1 ; X1 := XIN ;
15 ELSE XOUT := 0.0 ; X1 := XIN ; X2 := XIN ; X3 := XIN ;
16 END_IF ;
17 END_FUNCTION_BLOCK
18
19

```

После завершения редактирования, по нажатию пользователем кнопки *Компилировать*, будет выполнена трансляция кода *ST*. В нижнем поле пользователь в случае удачной компиляции получит сообщение - *Компиляция прошла успешно*, либо информации о возникших в результате компиляции ошибках.

```

Общие | Архив | Опрос | Код
1 FUNCTION_BLOCK Main_program
2 VAR_INPUT
3   RUN : BOL ; (* 0 = reset *)
4   XIN : REAL ; (* Input to be differentiated *)
5   CYCLE : TIME ; (* Sampling period *)
6 END_VAR
7 VAR_OUTPUT
8   XOUT : REL ; (* Differentiated output *)
9 END_VAR
10 VAR X1, X2, X3 : REAL ; END_VAR
11 I RUN THEN
12   XOUT := (3.0 * (XIN - X3) + X1 - X2)
13         / (10.0 * TIME_TO_REAL(CYCLE)) ;
14   X3 := X2 ; X2 := X1 ; X1 := XIN ;
15 ELSE XOUT := 0.0 ; X1 := XIN ; X2 := XIN ; X3 := XIN ;
16 END_IF ;
17 END_FUNCTION_BLOCK
18
19

```

Line	Description	File	Path
1	Program 'C:\Documents and Settings\пользователь\Рабочий стол\Проблема с сервером'...		
1	M0000 is already running!		
11	Declaration FUNCTION_BLOCK MAIN_PROGRAM (FUNCTION_BLOCK MAIN_PROGRAM VAR...	code.st	
11	Function block declaration: waiting for function block body (FUNCTION_BLOCK MAIN_PROG...	code.st	

Ошибки | Вывод | Стр. выполнения | Локальные переменные

0 Errors | 0 Warnings | 0 Messages

Применить | Отменить | Справка

Далее, в случае успешной компиляции, по нажатию кнопки *применить*, в дереве объектов у данного ФБ появятся входы и выходы, описанные у стартового функционального блока или программы.



Ниже приведено **Описание стандарта**.

В языке ST поддерживаны простые типы (логический, целочисленные, вещественные, битовые, строковые, времени), массивы, структуры, перечислимые типы и диапазоны.

Логический тип обозначается как *bool*. Может принимать два значения – *True* и *False*, *1* и *0*.

Битовые переменные обозначаются как *BYTE*, *WORD*, *DWORD*, *LWORD*.

Целочисленные переменные обозначаются *SINT*, *USINT*, *INT*, *UINT*, *DINT*, *UDINT*, *LINT*, *ULINT*.

Типы данных модуля ST и их аналоги в MasterSCADA.

Тип в ST	Кол-во байт	Верхний предел	Нижний предел	Аналог в MasterSCADA
BYTE	1	0	255	Беззнаковый короткий целый
WORD	2	0	65535	Беззнаковый короткий целый
DWORD	4	0	4294967295	Беззнаковый целый
LWORD	8	0	1,84467E+19	Вещественный дв. точности
SINT	1	-128	127	Целый
INT	2	-32768	32767	Целый
DINT	4	-2147483648	2147483647	Целый
LINT	8	-9,22337E+18	9,22337E+18	Вещественный дв. точности
USINT	1	0	255	Беззнаковый короткий целый
UINT	2	0	65535	Беззнаковый короткий целый
UDINT	4	0	4294967295	Беззнаковый целый
ULINT	8	0	1,84467E+19	Вещественный дв. точности

Из таблицы видно, что на один тип переменной MasterSCADA приходится несколько типов языка ST. Это нужно учитывать при объявлении переменных. Количество байт нужно учитывать при операциях сдвига.

Вещественные переменные – это переменные с плавающей запятой, в языке ST они обозначаются как *Real* и *LReal*.

Тип в ST	Кол-во байт	Аналог в MasterSCADA
Real	4	Вещественный
LReal	8	Вещественный дв. точности

Строковые переменные обозначаются *String*. Тип строка поддерживается как вход или выход ФБ ST только при работе на *hoste*.

Переменные типа **Время**:

- *TIME* – представляет собой длительность интервалов времени в миллисекундах;
- *TOD* – содержит время суток начиная от нуля часов;
- *DATE* – содержит календарную дату;
- *DT* – содержит время и дату.

Все типы времени на выходе ФБ в MasterSCADA представляются как тип *Время*. Таким образом основной тип переменных времени – *DT*. Формат *TIME* удобен для задания длительности при работе с таймерами.

Вещественным переменным можно присваивать значения целочисленных переменных. Во всех остальных случаях необходимо использовать специальные функции преобразования.

Массив – хранилище различных значений переменной одного типа. Массив может одномерным и многомерным.

Объявление массива: *Имя_массива: array[n1..v1,n2..v2,n3..v3] of тип_данных*, где *n1,n2,n3* – нижней предел индекса, *v1, v2, v3* – верхний предел индекса. Тип данных может быть любым, включая пользовательские структуры и массивы.

➤ **Пример:**

```
VAR
```

```
a:array[0..5] of real; (*одномерный массив вещественных чисел*)
```

```
END_VAR
```

Инициализировать массив можно при объявлении переменной: `a:array[0..5] of real:=1,8,10,16,42,8`. Либо используя индекс, обращаться к элементу массива:

```
a[0]:=0;
```

```
a[2]:=5;
```

В качестве номера индекса, можно использовать не только константу, но и целочисленную переменную:

```
for i:=0 to 5 do
```

```
  a[i]:=EXPT(i,2);
```

```
end_for;
```

Структура – это создаваемый пользователем новый тип данных, состоящих из других типов данных.

Объявление структуры:

```
TYPE имя_структуры:
```

```
  STRUCT
```

```
  Переменная1 : тип_переменной;
```

```
  Переменная2: тип_переменной;
```

```
  ....
```

```
  ПеременнаяN: тип_переменной;
```

```
  END_STRUCT;
```

```
END_TYPE
```

Для обращения к элементам структуры, нужно воспользоваться следующей инструкцией: `Имя_структуры.имя_переменной`.

- **Например:** создадим структуру описывающей человека, а именно – имя, фамилия, номер телефона, наличие работы.

```
TYPE TPerson:
```

```
  STRUCT
```

```
  Name: String; (*имя*)
```

```
  Surname:string; (*фамилия*)
```

```
  Number: dint; (*номер телефона*)
```

```
  Worker: bool; (*наличие работы*)
```

```
  END_STRUCT;
```

```
END_TYPE
```

Теперь несколькими переменными можно оперировать как с одной переменной – присваивать значения другой аналогичной переменной, создавать массивы. Чтобы присвоить значения нужно выполнить:

```
VAR
```

```
Person1:TPerson; (*объявление переменной*)
```

```
END_VAR
```

```
Person1.Name:="Виталий";
```

```
Person1.Surname:="Петров";
```

```
Person1.Number:=8901234;
```

```
Person1.Worker:=true;
```

Переменные могут быть *входными* (вход ФБ), *выходными* (выход ФБ), *входными и выходными* (вход и выход ФБ), *локальными*. Тип переменной задается в специальной секции.

При задании переменной используется следующий синтаксис: *Имя: тип*, где *Имя* – имя переменной, *тип* – тип данных, для хранения которых предназначена переменная. В имени переменной можно использовать любые буквы, цифры и знак подчеркивания *_*. Имя переменной не должно начинаться с цифры, и иметь пробелов.

➤ **Примеры объявления различных переменных:**

Входные:

```
VAR_INPUT
```

```
in:bool; (*логическая переменная*)
```

```
in1:int; (*целочисленная переменная*)
```

```
(*остальные входные переменные*)
```

```
END_VAR
```

Выходные:

```
VAR_OUTPUT
```

```
out:int; (*целочисленная переменная*)
```

```
(*остальные выходные переменные*)
```

```
END_VAR
```

Входные и выходные:

```
VAR_IN_OUT
```

```
in_out:real; (*вещественная переменная*)
```

```
(*остальные переменные*)
```

```
END_VAR
```

Локальные переменные:

```
var
```

```
i: int;
```

```
(*остальные локальные переменные*)
```

```
end_var
```

После *объявления*, с переменной можно производить различные действия – *присваивание*, *сравнение*, *математические операции*. Для проведения операции присваивания используется оператор *=*. Переменной можно присваивать числа (константы) или значения других совместимых переменных: *out:=1; out:=in*.

Операции условного выбора IF... THEN...ELSE.

Используя инструкцию *IF*, можно проверить условие, и в зависимости от истинности этого условия выполнить какие-либо действия.

Синтаксис:

```
IF <логическое_выражение> THEN
```

```
<Список_инструкций1>;
```

```
ELSE
```

```
<Список_инструкций2>;
```

```
END_IF;
```

Если логическое выражение истинно, то выполняется *Список_инструкций1*, в противном случае *Список_инструкций2*.

➤ **Пример:**

```

if (davl>5) then (*Если давление выше 5*)
klap:=false; (*Закреть клапан*)
else klap:=true; (*Иначе открыть клапан*)
end_if;

```

С помощью инструкции **CASE** можно сопоставить различным значениям переменной различные инструкции.

Синтаксис:

CASE <переменная> OF

<Значение_переменной1>: <Список_инструкций1>;

<Значение_переменной2>: <Список_инструкций2>;

<Значение_переменной3>, <Значение_переменной4>: <Список_инструкций2>;

<Значение_переменной5>..<Значение_переменной10>: <Список_инструкций2>;

.....

<Значение_переменной_i>: <Список_инструкций_i>;

ELSE <Список_инструкций_исключения>;

CASE работает следующим образом:

- Если <переменная> принимает значение равное <Значение_переменной_i> то выполняется <Список_инструкций_i>;
- Если <переменная> не соответствует ни одному из перечисленных значений, то выполняется <Список_инструкций_исключения>;
- Если необходимо чтобы один и тот же список инструкций выполнялся для различных значений переменной, то значения переменной нужно перечислить через запятую;
- Если необходимо чтобы один и тот же список инструкций выполнялся для диапазона переменных, необходимо отметить этот диапазон указав начальное и конечное значение диапазона разделив его двумя точками.

➤ **Пример:**

```

case in2 of (*Переменная*)
1: out:=1; (*Инструкции*)
out1:=2;
2: out:=2;
3,4: out:=3;
5..10: out:=4;
else out:=5;
end_case;

```

Цикл FOR можно использовать в том случае, если некоторую последовательность действий (инструкций программы) нужно выполнить определенное количество раз.

Синтаксис:

FOR <счетчик:=нач_знач> TO <конечн_знач> BY <величина шага> DO

<Список_инструкций >;

END_FOR;

FOR работает следующим образом. В начале, переменной <счетчик> присваивается значение равное <нач_знач>, затем выполняются инструкции внутри цикла. После каждого цикла <счетчик> увеличивается на значение равное <величина шага> и сравнивается с <конечн_знач>, если значение счетчика меньше конечного значения – цикл выполняется. Параметр BY <величина шага> - необязательный, в случае его отсутствия <счетчик> увеличивается на единицу.

➤ **Пример:**


```

for i:=0 to 10 BY 2 do
  out1:=out1+5;
end_for;

```

Цикл WHILE. Условием выхода из цикла *WHILE* является логическое выражение. Цикл *WHILE* используется в том случае, если некоторую последовательность действий (инструкций программы) надо выполнить несколько раз, причем необходимое число повторений во время разработки программы неизвестно, и может быть определено только во время исполнения программы. Примером использования цикла *WHILE* является поиск элементов в массиве.

Синтаксис:

```

WHILE <логическое_выражение> DO
< Список_инструкций >;
END_WHILE;

```

Инструкции внутри цикла *WHILE* выполняются до тех пор пока <логическое_выражение> является истинным. Если <логическое_выражение> еще до первого выполнения цикла имеет значение *false*, то инструкции не выполняются ни разу. Если <логическое_выражение> будет иметь неизменным значение *true*, то цикл будет выполняться бесконечно.

➤ **Пример:**

```

WHILE out1<10 DO
  out1:=out1+1;
end_while;

```

Цикл REPEAT. Цикл *REPEAT* схож с циклом *WHILE*, с одним лишь отличием – проверка условия выполнения инструкций осуществляется в конце цикла, а не в начале как у цикла *WHILE*. Таким образом цикл *REPEAT* выполняется как минимум один раз.

Синтаксис:

```

REPEAT
< Список_инструкций >;
UNTIL <логическое_выражение>
END_REPEAT;

```

Инструкции внутри цикла *REPEAT* выполняются до тех пор пока <логическое_выражение> является ложным. Таким образом <логическое_выражение> - это условие выхода из цикла.

➤ **Пример:**

```

REPEAT
  out1:=out1+1;
UNTIL out1>10
END_REPEAT;

```

Инструкция EXIT. Инструкцию *EXIT* можно использовать в циклах *FOR*, *WHILE*, *REPEAT* если необходимо прервать выполнения цикла.

➤ **Пример:**

```

REPEAT
  out1:=out1+1;
  if out1=8 then
    exit;
  end_if;
UNTIL out1<10
END_REPEAT;

```

MasterSCADA позволяет создавать собственные функции, функциональные блоки и структуры данных. Это позволяет сократить время разработки за счет внесения часто используемого кода в библиотеку. Это также упрощает разработку, отладку и модернизацию кода.

Функция - это простейший из создаваемых компонентов. Функция может иметь несколько входных значений и формировать на выходе одно выходное значение. При объявлении функции нужно указать тип возвращаемого значения, заканчивается объявление функции оператором *END_FUNCTION*. Например, функция возвращает вещественную переменную:

```
FUNCTION My_FUN:real;
```

```
<инструкции>
```

```
END_FUNCTION
```

Имя, данное функции используется как выходное значение – присвоенное ему в теле функции значение возвращается в вызываемый код. Входные переменные объявляются в стандартной секции *Var_input*:

```
VAR_INPUT
```

```
In: int;
```

```
In2:real;
```

```
END_VAR
```

Функция может содержать внутренние переменные, которые объявляются также как и при создании программы – при помощи инструкции *var ... end_var*.

- **Пример.** Создадим функцию, которая будет извлекать квадратный корень из одной входной переменной и складывать с другой входной переменной.

```
FUNCTION My_SQRT : real
```

```
(*входные переменные*)
```

```
VAR_INPUT
```

```
in:real; (*первая входная переменная*)
```

```
in1:int; (*вторая входная переменная*)
```

```
END_VAR
```

```
(*локальные переменные*)
```

```
VAR
```

```
k:real ; (*локальная переменная*)
```

```
END_VAR
```

```
(*КОД ФУНКЦИИ*)
```

```
k:=SQRT(in); (*вычисляем квадратный корень*)
```

```
My_SQRT:=in1+k; (*сложение со второй входной переменной и присвоение функции*)
```

```
END_FUNCTION
```

Для того чтобы использовать функцию в коде программы, ее нужно вызвать. Для вызова функции используется следующий синтаксис: *имя_функции(параметр1,параметр2,... параметрN)*, где, *Параметр* – значения, передаваемые в функцию.

- **Пример:**

```
VAR
```

```
first:real;
```

```
second:int;
```

```
END_VAR
```

```
VAR_OUTPUT
```

```
out:real;
```

```
END_VAR
```

```
first:=11.238; (*первая переменная в функции (real) - из нее извлекается квадратный корень*)
```

```
second:=2; (*вторая переменная (int) - с ней складывается вычисленное значения корня*)
```

```
out:=My_SQRT(first,second); (*порядок записи переменных соответствует порядку заданному при объявлении переменных VAR_INPUT*)
```

```
(*в результате в out=5.3523*)
```

В теле функции допускается вызывать другие функции.

Функциональный блок имеет неограниченное количество входных и выходных переменных, может иметь локальные переменные, а также вызывать функции. Для работы с функциональным блоком, в коде нужно объявить экземпляр блока и далее вести работу уже с ним. Экземпляры одного и того же блока работают независимо друг от друга.

Объявление функционального блока:

```
FUNCTION_BLOCK имя_блока
```

```
END_FUNCTION_BLOCK
```

Внутри блока нужно разместить объявление внутренних, внешних и локальных переменных. Создадим для примера функциональный блок, который будет использовать созданную ранее функцию *My_SQRT*, передавая в нее значения входов, а затем возведет полученное из функции значение в степень используя стандартную функцию *EXPT*.

➤ **Пример:**

```
FUNCTION_BLOCK My_FB
```

```
(*входные переменные*)
```

```
VAR_INPUT
```

```
in:real;
```

```
in1:int;
```

```
ste:int;
```

```
END_VAR
```

```
(*выходные переменные*)
```

```
VAR_OUTPUT
```

```
output:real;
```

```
END_VAR
```

```
(*локальные переменные*)
```

```
VAR
```

```
p:real;
```

```
END_VAR
```

```
(*Код функционального блока*)
```

```
p:=My_SQRT(in,in1); (*вызов функции*)
```

```
output:=EXPT(p,ste); (*возведение в степень*)
```

```
END_FUNCTION_BLOCK
```

Большинство функциональных блоков вызываются следующим образом (на примере блока счетчика *CTU*). Сначала, в секции объявления переменных, нужно объявить экземпляр функционального блока (подобно объявлению обычной переменной):

```
var
```

```
Count:CTU; (*переменная Count - тип счетчик*)
```

```
end_var
```

Функциональный блок может иметь любое количество экземпляров, каждый из которых будет работать независимо от других. Далее в разделе кода, происходит вызов функционального блока, и работа с его входами и выходами:

Count(CU:=in,R:=in2,PV:=10,Q=>out1,CV=>out2), где:

- *Count* – экземпляр функционального блока типа *CTU*;
- *CU,R,PV* – входные переменные блока;
- *Q,CV* – выходные переменные блока.

Входным переменным блока присваивание значений осуществляется при помощи инструкции =, получение выходных переменных блока осуществляется при помощи инструкции =>. Можно получить доступ к входам и выходам экземпляра функционального блока используя инструкцию:

Имя_экземпляра.имя_переменной

Используя такую инструкцию можно по другому представить аналогичный код:

Count.CU:=in;

Count.R:=in1;

Count();

out1:=Count.Q;

out2:=Count.CV;

При подобном методе присваивания входных переменных блока, распространенной ошибкой является отсутствие в коде инструкции: Count(). Если данная инструкция отсутствует то блок не будет вызываться, а значит и не будет выполняться.

- ❖ **Важно!** *Некоторые функциональные блоки будут работать, только если установлен периодический опрос ФБ ST. Например, блоки таймеров и детекторов импульсов.*

Ниже представлена таблица стандартных функций:

Операторы	Расшифровка
Операторы сравнения	GT - убывающая последовательность. GE - монотонная последовательность. EQ – равенство. LE - монотонная последовательность. LT - возрастающая последовательность.
Арифметические операторы	ADD - оператор сложения. MUL - оператор умножения. SUB - оператор вычитания. DIV - оператор деления. MOD - вычисление остатка от деления целочисленных значений. ABS - модуль числа. SQRT - извлечение квадратного корня. LN - вычисление натурального логарифма. EXP - вычисление экспоненты. SIN - вычисление синуса. COS - вычисление косинуса. TAN - вычисление тангенса. ASIN - вычисление арксинуса. ACOS - вычисление арккосинуса. ATAN - вычисление арктангенса. EXPT - вычисление числа в степени.
Логические и битовые операторы	AND – оператор И. OR - оператор ИЛИ. NOT - оператор НЕ. XOR - оператор Исключающее ИЛИ.
Операторы выборки	SEL - бинарный выбор. MAX - наибольшее из двух значений. MIN - наименьшее из двух значений. MUX - расширяемый мультиплексор. LIMIT - функция ограничения.
Строковые функции	LEN - возвращает длину строки.

Операторы	Расшифровка
	LEFT - возвращает левую часть строки заданной длины. RIGHT - возвращает правую часть строки заданной длины. MID - выделение фрагмента строки. CONCAT - объединение строки. INSERT - позволяет вставить одну строку в определённую позицию другой строки. DELETE - удаление части строки. REPLACE - замена части подстроки другой строкой с указанной позиции. FIND - поиск подстроки в строке.
Операции преобразования	REAL_TO_INT - преобразование вещественной переменной в целочисленную. TIME_TO_REAL - преобразование переменной времени в вещественную.
Функции для работы со временем	ADD_TIME или ADD - суммирует две переменные <i>TIME</i> . ADD_TOD_TIM или ADD - прибавляет к переменной типа <i>TOD</i> (время) значение переменной типа <i>TIME</i> . SUB_TIME или SUB - разность двух переменных <i>TIME</i> . SUB_DATE_DATE или SUB - разность двух переменных типа <i>DATE</i> . SUB_TOD_TIME или SUB - вычитает из переменной типа <i>TOD</i> (время) значение переменной <i>TIME</i> . SUB_TOD_TOD или SUB - разность двух переменных <i>TOD</i> (время). SUB_DT_TIME или SUB - вычитает из переменной <i>DT</i> значение переменной <i>TIME</i> . SUB_DT_DT или SUB - разность двух переменных <i>DT</i> . MULTIME или MUL - умножает переменную типа <i>TIME</i> на любую числовую переменную. DIVTIME или DIVE - делит переменную типа <i>TIME</i> на любую числовую переменную. CONCAT_DATE_TOD - объединяет значения переменных <i>DATE</i> и <i>TOD</i> в одну переменную типа <i>DT</i> .
Операторы сдвига	SHL - побитный сдвиг влево. SHR - побитный сдвиг вправо. ROL - циклический сдвиг влево. ROR - циклический сдвиг вправо.

Подробная информация по каждому из операторов содержится в справке MasterSCADA.


Стандартные функциональные блоки описаны в таблице ниже.

Функциональный блок	Расшифровка
Триггеры	RS - триггер с доминантой выключения. SR - триггер с доминантой включения.
Детекторы импульсов	R_TRIG - генерирует импульс продолжительностью один цикл по переднему фронту входного сигнала. F_TRIG - генерирует импульс продолжительностью один цикл по заднему фронту входного сигнала.
Счётчики	CTU (CTU_DINT, CTU_LINT, CTU_UDINT, CTU_ULINT) - реализует функцию инкрементального счётчика. CTD (CTD_DINT, CTD_LINT, CTD_UDINT, CTD_ULINT) - реализует функцию декрементального счётчика. STUD (STUD_DINT, STUD_LINT, CTD_ULINT) - реализует функцию реверсивного счётчика.
Таймеры	TP - таймер. TON - таймер с задержкой включения. TOF - таймер с задержкой выключения.

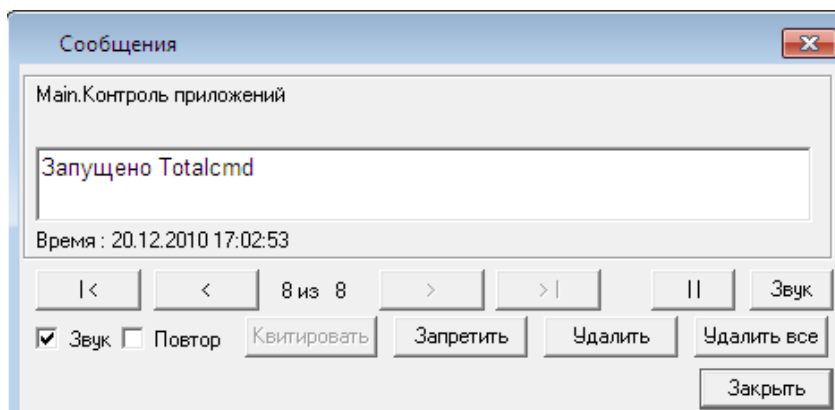
Подробная информация по каждому из функциональных блоков языка ST содержится в справке MasterSCADA.

Контроль приложений

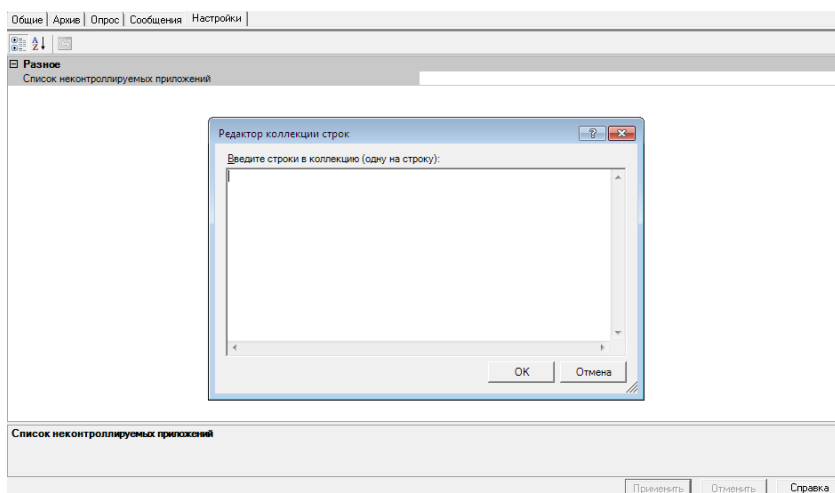
Вид функционального блока в дереве объекта:

 Контроль приложений

На каждом цикле опроса запрашивается список приложений, имеющих окна и доступных через панель задач Windows. По каждому запущенному приложению выдаётся сообщение: *Запущено '<имя приложения>'*. По каждому закрытому (которое работало на предыдущем опросе) - *Закрыто '<имя приложения>'*.



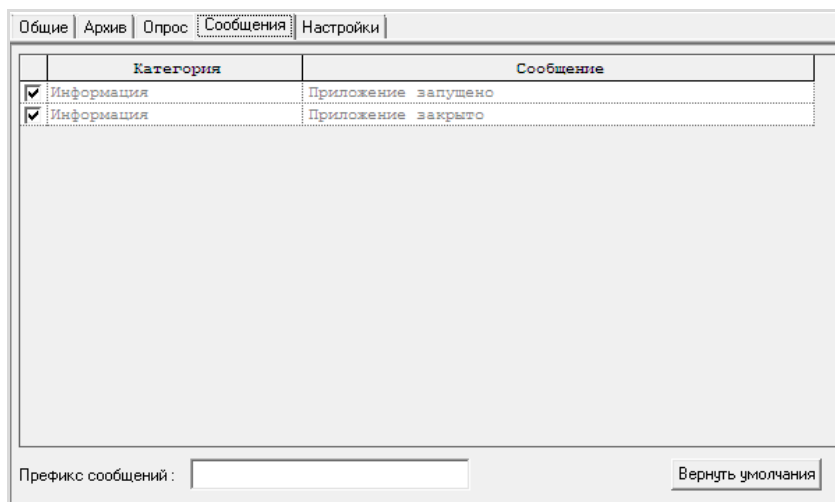
Страница Настройки ФБ Контроль приложений выглядит следующим образом:



Список неконтролируемых приложений - список имен приложений, о запуске которых не нужно информировать:

Сообщения ФБ:

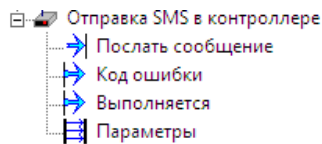
- Приложение запущено (категория - Информация);
- Приложение закрыто (категория - Информация).



Отправка смс в контроллере

Данный функциональный блок служит для отправки SMS с модема, который подключен к контроллеру.

Вид функционального блока в дереве объекта:



В настройках контроллера в дереве системы у порта контроллера, к которому подключен модем, необходимо указать интерфейс *Модем*. Если ранее через этот модем происходила связь с верхним уровнем, то отправка SMS будет осуществляться через этот же модем и не потребует дополнительных настроек в дереве системы. Если модем предполагается использовать только для отправки SMS, необходимо указать только интерфейс *Модем*, не указывая протокол. Если протокол не выбран, то номер телефона не указывается.

Параметры интерфейса	
Номер порта	1
Скорость	9600
Номер телефона	
Строка инициализации	ATE0Q0V1X4&C0&D2&S1S0=
Пароль	20816

Протокол: Не выбран

Обратите внимание на форму записи телефона *Получателя*.

Общие	
Время ожидания	10000
Макс. число команд	2
Получатель	+79xxxxxxxx
Шаблон сообщения	
Разное	
Количество параметров	1

Описание шаблона сообщения. Шаблон состоит из статического текста с включением значений параметров. В статическом тексте можно использовать следующие символы:

- $\backslash r$ – $\langle CR \rangle$ - возврат каретки (код $0D$);
- $\backslash n$ - $\langle LF \rangle$ - перевод строки;
- $\backslash \backslash$ - символ \backslash

Значения входов ФБ в запросах. Общий вид:

$\{P(\langle \text{Индекс_вх} \rangle): \langle \text{Длина_в_кадре} \rangle: \langle \text{Формат_данных} \rangle\}$

$\{P(\langle \text{Нач_вх} \rangle - \langle \text{Кон_вх} \rangle): \langle \text{Длина_в_кадре} \rangle: \langle \text{Формат_данных} \rangle\}$

В данных запросах:

- $\langle \text{Индекс_вх} \rangle$ - это индекс входа ФБ в группе, индексирование начинается с 0;
- $\langle \text{Нач_вх} \rangle$ и $\langle \text{Кон_вх} \rangle$ - это индексы начального и конечного входов ФБ в группе, индексирование начинается с 0 (если нужно вставить значения нескольких входов подряд);
- $\langle \text{Длина_в_кадре} \rangle$ – минимальное количество символов, которое занимает одно передаваемое значение (если значение меньше, то оно слева дополняется нулями). Для вещественных значений через точку можно указать количество знаков после запятой;
- $\langle \text{Формат_данных} \rangle$:
 - d – означает, что символы в кадре представляют собой число в десятичной системе;
 - h - символы в кадре представляют собой число в шестнадцатеричной системе, цифры $a-f$ в нижнем регистре;
 - H - символы в кадре представляют собой число в шестнадцатеричной системе, цифры $A-F$ в верхнем регистре;
 - f – символы в кадре представляют десятичное вещественное число с точкой, отделяющей дробную часть.

Нечеткая логика

Нечеткое И

Значения переменных на *Входах* масштабируются в интервале от 0 до 1 в соответствии с выбранным диапазоном изменения. Если диапазон изменения переменной выбран не был, то он по умолчанию устанавливается равным 0...100. На *Выход* подается значение равное $\min(\text{Вход1}, \text{Вход2})$. Вид функционального блока в дереве объекта:



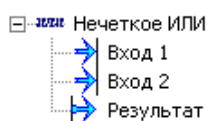
Входы и входы:

	Обозначение	Тип
Входы	Вход1	Вещественный дв. точности
	Вход2	Вещественный дв. точности
Выход	Результат	Вещественный дв. точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Нечеткое ИЛИ

Значения переменных на *Входах* масштабируются в интервале от 0 до 1 в соответствии с выбранным диапазоном изменения. Если диапазон изменения переменной выбран не был, то он по умолчанию устанавливается равным 0...100. На *Выход* подается значение равное $\max(\text{Вход1}, \text{Вход2})$. Вид функционального блока в дереве объекта:



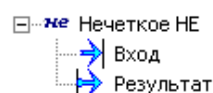
Входы и входы:

	Обозначение	Тип
Входы	Вход1	Вещественный дв. точности
	Вход2	Вещественный дв. точности
Выход	Результат	Вещественный дв. точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Нечеткое НЕ

Значение переменной на *Входе* масштабируется в интервале от 0 до 1 в соответствии с выбранным диапазоном изменения. Если диапазон изменения переменной выбран не был, то он по умолчанию устанавливается равным 0...100. На *Выход* подается разность 1 и значения на *Входе*. Вид функционального блока в дереве объекта:



Входы и входы:

	Обозначение	Тип
Вход	Вход	Вещественный дв. точности
Выход	Результат	Вещественный дв. точности

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Другие

Аналоговый ПИД-регулятор

Функциональный блок *Аналоговый ПИД-регулятор* осуществляет автоматическое управление аналоговым исполнительным механизмом:

- По внутреннему ПИД-закону регулирования;
- Дистанционно, позволяя непосредственно оператору задавать управляющее воздействие.

Имеется возможность использовать его в каскадной схеме регулирования. ФБ *Аналоговый ПИД-регулятор* предоставляет возможность работы в 4-х режимах работы: *Дистанция*, *Автомат*, *Насквозь*, *Каскад*. Каждому из режимов характерны следующие свойства:

- *Дистанция* - управляющее воздействие задается оператором вручную в окне управления регулятором;
- *Автомат* - управляющее воздействие рассчитывается внутри ФБ по ПИД – закону регулирования. Задание устанавливается оператором вручную из окна управления;
- *Каскад* - управляющее воздействие рассчитывается внутри ФБ по ПИД – закону регулирования. Задание устанавливается внешней командой по входу *Задание*;
- *Насквозь* - Управляющее воздействие рассчитывается внешней логикой. Управление передается на выход ФБ со входа *Насквозь*;

Приоритет переключения между режимами соответствует порядку описания режимов в сторону уменьшения приоритета. ПИД закон регулирования, реализованный в данном ФБ, описывается формулой в операторах Лапласа.

В рамках реализации данного функционального блока внесена функция фильтрации входного параметра. Она реализована по апериодическому закону и представляет собой фильтр первого порядка. В алгоритм функционирования ФБ внесена процедура обеспечения безударного перехода из одного режима в другой (статическая балансировка по выходу). Например, при переходе из режимов *Дистанция* или *Насквозь* в режимы *Автомат* или *Каскад*. В функциональном блоке предусмотрена возможность диагностики процесса регулирования. Аварийный сигнал формируется в случае, когда за заданное контрольное время величина рассогласования не стала меньше требуемой логикой работы алгоритма.

В алгоритме регулирования имеется возможность настройки зоны нечувствительности. Есть возможность выбора их двух типов, которые описаны соответствующими формулами:

- $y=0$, если $|x| \leq$ зона нечувствительности, $y=x$, если $|x| >$ зона нечувствительности
- $y=0$, если $|x| \leq$ зона нечувствительности, $y=(|x|-\delta) \cdot \text{sign}(x)$, если $|x| >$ зона нечувствительности

Функциональность работы ПИД-регулятора расширена введением в алгоритм расчета оптимального управляющего воздействия двух корректирующих коэффициентов – аддитивного и мультипликативного. Значение *аддитивного коэффициента* прибавляется к величине рассогласования. Это позволяет учитывать в регулировании дополнительные опережающие сигналы для выработки более скорой реакции на возмущения. При использовании *мультипликативной составляющей*, скорректированное значение величины рассогласования умножается на число, указанное в настройках ФБ (используется в парных потоках при выходе из строя одного из потоков).

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

ВходыФБ:

Имя	Тип	Примечание
Параметр	Вещественный дв. точности	Величина регулируемого параметра.
Задание	Вещественный дв. точности	Величина задания для режима <i>Каскад</i> .
Аддитивная коррекция	Вещественный дв. точности	Это значение прибавляется к величине рассогласования. Используется для введения в САР дополнительных управляющих сигналов.
Запрет Больше	Логический	Запрет изменения величины управления в сторону <i>Больше</i> в режимах <i>Автомат</i> или <i>Каскад</i> .

Имя	Тип	Примечание
Запрет Меньше	Логический	Запрет изменения величины управления в сторону <i>Меньше</i> в режимах <i>Автомат</i> или <i>Каскад</i> .
Использовать мультипликативную коррекцию	Логический	При поступлении сигнала на данный вход, скорректированное значение величины рассогласования умножается на коэффициент, введенный в свойствах настроек ФБ в строке <i>Мультипликативный коэффициент</i> .
Вкл Дистанция	Логический	При включенном режиме управляющее воздействие задается оператором вручную в окне управления регулятором.
Вкл Автомат	Логический	Управляющее воздействие рассчитывается внутри ФБ по ПИД – закону регулирования. Задание устанавливается оператором вручную из окна управления.
Вкл Каскад	Логический	Управляющее воздействие рассчитывается внутри ФБ по ПИД – закону регулирования. Задание устанавливается внешней командой по входу <i>Задание</i> .
Вкл Сквозной	Логический	Управляющее воздействие рассчитывается внешней логикой. Управление передается на выход ФБ со входа <i>Насквозь</i> .
Насквозь	Вещественный дв. точности	Величина управляющего воздействия, передаваемого на выход ФБ. Используется при режиме <i>Сквозной</i> .
Авария	Логический	Внешний аварийный сигнал (резервный), используемый разработчиком.
Квитирование аварии	Логический	Внешняя команда квитирования ошибок.
Балансировка	Логический	При поступлении на данный вход сигнала <i>Истина</i> происходит статическая балансировка управляющего воздействия по входу <i>Параметр балансировки</i> . Используется в каскадных схемах, когда ведомый контур переводится в <i>Дистанционный режим</i> или находится в аварийном состоянии, что позволяет ликвидировать <i>разбег</i> между контурами регулирования в нештатных ситуациях.
Параметр балансировки	Вещественный дв. точности	Значение параметра балансировки присваивается внутреннему интегрирующему звену (которое подается на выход <i>Управление</i>) при подаче <i>Истина</i> на вход <i>Балансировка</i> .
Kp	Вещественный дв. точности	Коэффициент пропорциональной составляющей регулятора.
Td	Вещественный дв. точности	Постоянная времени дифференцирования. Если равен 0, дифференциальная составляющая отключается.
Ti	Вещественный дв. точности	Постоянная времени интегрирования. Если равно 0, то интегральная составляющая отключается.

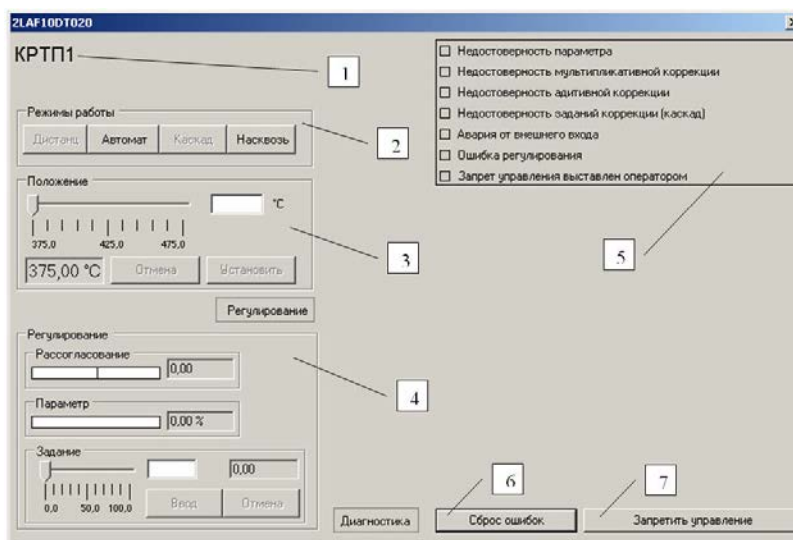
Входы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-байтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма и для данного ФБ не используется (всегда определяется неопределенным состоянием 8). Второй байт представляет состояние режима механизма (представим в двоичном виде): 0x00000001 – дистанция; 0x00000010 – автомат; 0x00000100 – каскад; 0x00001000 – аварийный переход в дистанцию; 0x00010000 – насквозь. Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000100 – квитирование аварии; 0x00010000 – неблокирующая авария (от внешнего входа, недоверие параметра). Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте - 262144) авария (пятый бит в третьем байте - 1048576) в режиме аварийный переход в дистанцию (четвертый бит во втором байте - 2048). На выходе будет (262144+1048576+2048+8) 1312776.
Выход Авария	Логический	Обобщенный сигнал <i>Авария</i> о возникших ошибках в работе ФБ
Запрет управления	Логический	Блокирует изменение состояния блока и в том числе

Имя	Тип	Примечание
Управление	Вещественный дв. точности	управляющего воздействия. Эта команда обладает наивысшим приоритетом. Выход регулятора (управляющее воздействие на исполнительный механизм). Диапазон изменения значений зависит от настроек <i>Верхний предел управляющего воздействия</i> и <i>Нижний предел управляющего воздействия</i> .
Каскад	Логический	Флаг, сигнализирующий о переходе работы ФБ в режим <i>Каскад</i> . Формируется внутренней логикой работы ФБ.
Автомат	Логический	Флаг, сигнализирующий о переходе работы ФБ в режим <i>Автомат</i> . Формируется внутренней логикой работы ФБ.
Дистанция	Логический	Флаг, сигнализирующий о переходе работы ФБ в режим <i>Дистанция</i> . Формируется внутренней логикой работы ФБ.
Сквозной	Логический	Флаг, сигнализирующий о переходе работы ФБ в режим <i>Сквозной</i> . Формируется внутренней логикой работы ФБ.
Аварийный переход в дистанцию	Логический	Флаг перехода в режим <i>Дистанция</i> при возникновении сигнала <i>Авария</i> .

Символ динамического элемента *Аналоговый ПИД-регулятор* состоит из прямоугольной подложки серого цвета, на которой расположены два элемента – круг, отображающий текущее состояние/режим регулятора, и надпись (сокращенное название регулятора, например, КРТП1 – корректирующий регулятор температуры пара). Текст надписи задается в редакторе мнемосхем для каждого экземпляра контрола отдельно.

Элемент динамический. При клике вызывается окно управления:



В окне управления оператору предоставляется возможность произвести следующие действия:

- Краткий технологический код и/или полное наименование регулятора;
- Кнопки задания режима работы регулятора;
- Слайдер позволяет вручную мышкой задать величину управляющего воздействия в режиме *Дистанция*. Для задания управляющего воздействия с клавиатуры имеется рядом специальное окно. Для подтверждения введенного управляющего воздействия с клавиатуры или мышкой присутствует кнопка *Установить*. Кнопка *Отмена* отменяет все управляющие воздействия от оператора;
- Блок регулирования, в котором оператор имеет возможность устанавливать значение задания и отслеживать величину параметра. После установления нового значения задания необходимо нажать кнопку *Ввод*. Нажатие клавиши *Отмена* приводит к возврату старого значения задания.
- Панель диагностики состояния регулятора. Вызывается при нажатии кнопки *Диагностика*.
- По нажатию на кнопку *Сброс ошибок* производится сброс мигания, регулятор переводится в режим *Дистанция* если находился в аварийном состоянии, сбрасывается ошибка регулирования;
- По нажатию на кнопку *Запретить управление* происходит блокировка изменения управляющего воздействия.

При нажатии на кнопку *Диагностика* появляется панель диагностики. Она содержит следующие индикаторы:

- Поступления плохого признака качества входного параметра;
- Недостоверность мультипликативной коррекции – формируется в момент поступления плохого признака качества на вход ФБ *Использовать мультипликативную коррекцию*;
- Недостоверность аддитивной коррекции – формируется в момент поступления плохого признака качества на вход ФБ *Аддитивная коррекция*;
- Недостоверность задания коррекции (каскад) – формируется в момент поступления плохого признака качества входного параметра;
- Авария от внешнего входа – формируется при поступлении внешнего аварийного сигнала, используемого разработчиком проекта;
- Ошибка регулирования – формируется в момент возникновения ошибки регулирования (при условии установки флага *Формировать аварию при ошибке регулирования* в лог.1);
- Запрет управления выставлен оператором – формируется при нажатии на кнопку *Запретить управление*;

Страница свойств *Настройки* ФБ *ПИД – регулятор аналоговый* выглядит следующим образом:

Общие Архив Опрос Сообщения Настройки Права доступа	
1 Настройка управляющего воздействия	
1.1 Верхний предел управляющего воздействия	100
1.2 Нижний предел управляющего воздействия	0
2 Режимы	
2.1 Использовать Автомат	Истина
2.2 Использовать Каскад	Истина
2.3 Использовать Насквозь	Ложь
3 Регулирование	
3.1 Верхний предел регулирования (ед. параметра)	100
3.2 Нижний предел регулирования (ед. параметра)	0
3.3 Максимальное значение рассогласования (%)	100
3.4 Балансировка задания регулятора	Не используется
3.5 Время динамической балансировки (с)	10
3.6 Использовать аддитивную коррекцию	Истина
3.7 Использовать мультипликативную коррекцию	Истина
3.8 Мультипликативный коэффициент	1
3.9 Максимальная скорость изменения рассогласования (ед.п.	0
4 Обработка параметра	
4.1 Постоянная времени фильтра параметра (с)	0
4.2 Тип зоны нечувствительности	Тип 2
4.3 Величина полузоны нечувствительности (ед. параметра)	0
5 Диагностика	
5.1 Выдержка при недостоверности параметра (с)	1
5.2 Диагностировать ошибку регулирования	Ложь
1.1 Верхний предел управляющего воздействия Верхний предел управляющего воздействия	
Применить Отменить Справка	

Функции элементов управления описаны в таблице ниже:

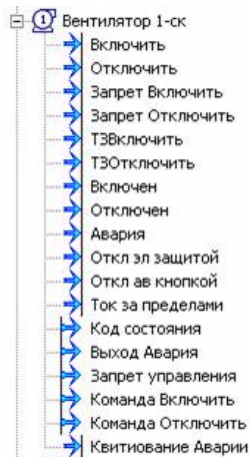
Элемент управления	Описание
Группа <i>Настройка управляющего воздействия</i>	Верхний предел управляющего воздействия - ограничение величины управляющего воздействия; Нижний предел управляющего воздействия -ограничение величины управляющего;
Группа <i>Режимы</i>	Использовать Автомат - использовать режим <i>Автомат</i> ; Использовать Каскад - использовать режим <i>Каскад</i> ; Использовать Насквозь -использовать режим <i>Насквозь</i> ;
Группа <i>Регулирование</i>	Верхний предел регулирования - ограничение величины задатчика; Нижний предел регулирования -ограничение величины задатчика; Максимальное значение рассогласование (%) - максимальное по абсолютному значению рассогласование, пропускаемое на регулятор. Если рассогласование превышает это значение, то рассогласование принимается равным этому значению;

Элемент управления	Описание
	<p>Время динамической балансировки, (с); Использовать аддитивную коррекцию - при включенном флаге, значение, поступающее на вход <i>аддитивная коррекция</i>, прибавляется к величине рассогласования; Использовать мультипликативную коррекцию -при включенном флаге позволяет использовать вход <i>Использовать Мультипликативную Коррекцию</i>. При подаче сигнала <i>Истина</i> на этот вход, значение величины рассогласования, с учетом аддитивной коррекции, умножается на коэффициент, введенный в свойствах настроек ФБ в строке <i>Мультипликативный коэффициент</i>; Мультипликативный коэффициент - тип значения - вещественный; Максимальная скорость изменения рассогласования.</p>
Группа <i>Обработка параметра</i>	<p>Постоянная времени фильтра параметра; Тип зоны нечувствительности - в настройке указывается, какая из функций используется при пересчете величины рассогласования.</p>
Группа <i>Диагностика</i>	<p>Выдержка при недостоверности параметра, с; Диагностировать ошибку регулирования - определяет наличие контроля правильности регулирования.</p>

Вентилятор односкоростной

Функциональный блок *Вентилятор односкоростной*, загружаемый в контроллер, осуществляет функцию управления односкоростным исполнительным механизмом, а также предоставляет возможность отображения на видеокadre мнемонического символа состояния исполнительного механизма, привязанного к этому ФБ.

Вид функционального блока в дереве объекта:



При поступлении команды на вход *Включить* происходит включение ИМ, если механизм был выключен. При поступлении команды на вход *Отключить* происходит отключение механизма. Переключения осуществляются в соответствии со следующей логикой:

- При поступлении команды на вход *Включить* устанавливается флаг *Идет переключение*. Проверяется состояние входа *Включен*. Если ИМ не включен, то выдается команда *Включить* и, по истечении времени T_d (Контрольное время включения/отключения), проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Не включился за контрольное время* и сбрасывается признак *Идет переключение* (мигание зеленого кружка внутри мнемосимвола). При успешном выполнении команды (наличие сигнала *Включен*) сбрасывается признак *Идет переключение*;
- Поступление команды на вход *Отключить* приводит к отключению механизма. Действия, выполняемые по команде от входа *Включить*, отменяются. Проверяется состояние входа *Включен*. При наличии включенного состояния выдается команда на его отключение и, по истечении времени T_d (Контрольное время включения/отключения), проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Неисполнение команды отключения*. При успешном выполнении команды (наличие сигнала *Отключен*) сбрасывается признак *Идет переключение*;

- Выставление признака *Запрет управления* через окно оператора, блокирует выдачу каких-либо команд из функционального блока. Эта команда обладает наивысшим приоритетом;
- Диагностика. Появление следующих диагностических сигналов блокирует команды по входам включения/отключения до квитирования - *Неисполнение команды включения/отключения, Отключен электрической защитой*. При наличии сигнала *Ток за пределами* блокируются только команды на включение ИМ. Контролируются:
 - Исполнение выданных команд на включение/отключение;
 - Факт срабатывания *Электрической защиты* по наличию сигнала *Истина* на входе *Отключён электрической защитой*; При квитировании данного события, ФБ посылает импульс на отключения для сброса электрической защиты;
 - Факт отключения ИМ Аварийной кнопкой по наличию сигнала *Истина* на входе *Отключен ав кнопкой*;
 - Факт выхода тока Вентилятора за допустимые пределы значения, по наличию сигнала *Истина* на входе *Ток за пределами*;
 - Несанкционированный доступ – в случае изменения состояния *включен/отключен* без команд от функционального блока и при отсутствии электрических защит;
 - Возможен контроль других аварийных событий не учтенных в данном ФБ по наличию сигнала *Истина* на входе *Авария*;
- Все команды, приходящие на ФБ, разделяют на группы приоритетов. Более приоритетные команды и блокировки блокируют менее приоритетные команды. Команды с одним приоритетом, но противоположные по направлению дополняют друг друга. Общий перечень команд в порядке увеличения приоритета:
 - Команды *Отключить* и Выключить от оператора. команды Включить и *Отключить* от внешней логики;
 - Команды *Запрет Включить* и *Запрет Отключить* от внешней логики;
 - Команды от технологических защит, подаваемые через входы *ТЗОтключить* и *ТЗВключить*;
 - Блокировки при обнаружении аварийного состояния;
 - Запрет управления выставленной оператором через окно управления.
- Выходные команды формируются в виде импульса заданной длительности (в настройках ФБ). Одновременная выдача противоположных команд блокируется.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - не требуется.

Входы ФБ:








Имя	Тип	Примечание
Включить	Логический	Служит для передачи команды от внешней логики на включение ИМ.
Отключить	Логический	Служит для передачи команды от внешней логики на отключение ИМ.
Запрет Включить	Логический	Служит для блокировки команд <i>Включить</i> от внешней логики и от оператора. Не распространяется на команды от технологических защит.
Запрет Отключить	Логический	Служит для блокировки команд <i>Отключить</i> от внешней логики и от оператора. Не распространяется на команды от технологических защит.
ТЗВключить	Логический	Служит для передачи команды от технологических защит на включение. Данная команда имеет повышенный приоритет и не блокируется входом <i>Запрет Включить</i> . Может быть заблокирована только запретом на управление выставленным лично оператором, либо аварийным состоянием ИМ.
ТЗОтключить	Логический	Служит для передачи команды от технологических защит на включение. Данная команда имеет повышенный приоритет и не блокируется входом <i>Запрет Включить</i> . Может быть заблокирована только запретом на управление выставленным лично оператором, либо аварийным состоянием ИМ.
Включен	Логический	Информационный вход. Служит для приема сигнала, подтверждающего факт включения вентилятора. Участвует в формировании обобщенного сигнала состояния ИМ.
Отключен	Логический	Информационный вход. Служит для приема сигнала, подтверждающего факт отключения вентилятора. Участвует в формировании обобщенного сигнала состояния ИМ.
Отключение эл. защитой	Логический	Информационно-аварийный вход. Служит для приема сигнала об отключении ИМ электрической защитой. Участвует в формировании обобщенного сигнала аварийного состояния ИМ. Данный сиг-

Имя	Тип	Примечание
		нал обычно фиксируется самой защитой и блокирует все попытки повторного включения ИМ, до тех пор, пока не будет сброшен оператором (через окно диагностики ФБ) или внешней логикой (через логический вход <i>Квитирование Аварии</i>) При квитировании данного сигнала функциональный блок выдает импульс на выходе <i>Команды Отключить</i> .
Отключение ав. кнопкой	Логический	Информационно-аварийный вход. Служит для приема сигнала об отключении ИМ аварийной кнопкой по месту. Участвует в формировании обобщенного сигнала аварийного состояния ИМ, которое в свою очередь требует квитирования со стороны оператора.
Авария	Логический	Информационно-аварийный вход. Служит для формирования других аварийных ситуаций связанных с данными ИМ, которые не были учтены разработчиками ФБ. Блокирует команды на включение или отключение ИМ.
Ток за пределами	Логический	Информационно-аварийный вход. Служит для приема сигнала о выходе тока за допустимые пределы. Участвует в формировании обобщенного сигнала аварийного состояния ИМ. При наличии данного сигнала блокируются все команды на включение ИМ.
Квитирование Аварии	Логический	Служит для передачи от внешней логики на квитирование аварийной ситуации в ИМ. Обычно используется для организации общего квитирования группы ИМ и параметров, размещенных, например, на одной мнемосхеме.

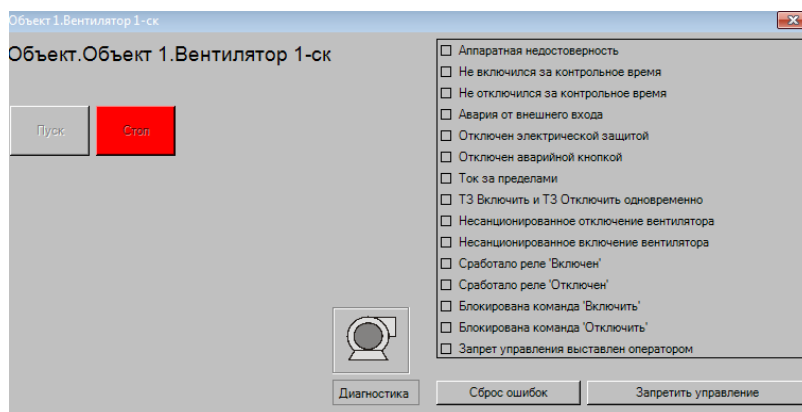
Выходы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-хбайтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма: 1 – включен; 2 – отключен; 3 – включается; 4 – отключается; 8 – недостоверное состояние. Второй байт представляет состояние режима механизма и для данного ФБ не используется. Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000010 – авария (от внешнего входа, не включился за контрольное время); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте -bin00000100 00000000 00000000 -> hex:040000 -> Dec:262144) авария (второй бит в третьем байте - bin:00000010 00000000 00000000 -> hex:020000 -> Dec:131072) при включенном состоянии вентилятора(hex:000001 -> Dec:1) . На выходе будет 262144+131072+1=393217 (hex:040000+020000+ 000001 = hex:060001).
Выход Авария	Логический	Обобщенный сигнал <i>Авария</i> о возникших ошибках в работе ИМ
Запрет управления	Логический	Информационный сигнал о выставленном оператором запрете на управление.
Команда Включить	Логический	Результатирующая команда <i>Включить</i> , выдаваемая на исполнение во внешние управляющие цепи.
Команда Отключить	Логический	Результатирующая команда <i>Отключить</i> , выдаваемая на исполнение во внешние управляющие цепи.

Мнемоническое отображение ФБ на мнемосхеме:

Отображение	Состояние	Поведение
	Вентилятор <i>Включен</i>	Ровный зеленый цвет
	Вентилятор <i>Отключен</i>	Ровный белый цвет
	Присутствует сигнал аварии	Мигает красная рамочка.
	Недостоверное состояние	Ровный серый цвет
	<i>Включен</i> и запрет управления	Запрещены все операции, включая управление от защит и блокировок
	<i>Отключен</i> и запрет управления	Запрещены все операции, включая управление от защит и блокировок
	Недостоверное состояние и запрет управления	Запрещены все операции, включая управление от защит и блокировок

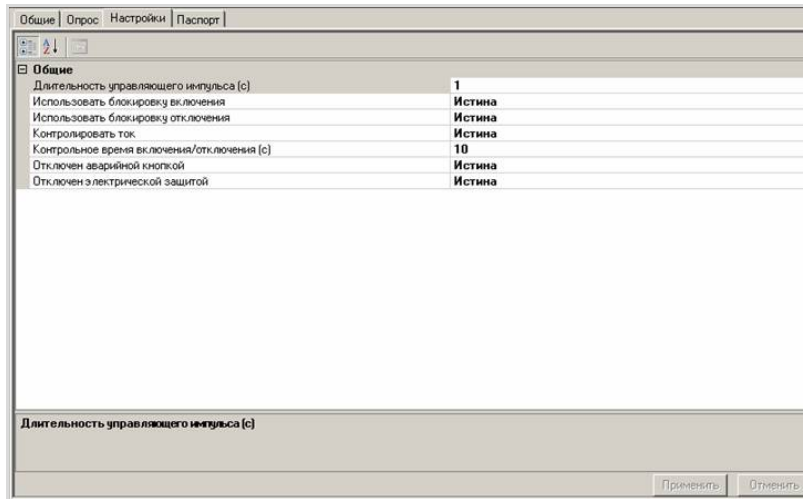
Элемент динамический, при клике вызывается окно управления:



По кнопке *Диагностика* вызывается панель состояния диагностики выбранного механизма. *Панель содержит индикаторы:*

- *Аппаратная неадекватность* - входные сигналы, отображающие состояние электродвигателя, имеют плохой признак качества;
- *Не включился за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Включен* по истечении контрольного времени с момента выдачи на выход команды *Включить*;
- *Не отключился за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Отключен* по истечении контрольного времени с момента выдачи на выход команды *Отключить*;
- *Авария от внешнего входа* - формируется при наличии тогда когда на входе *Авария* установлен и держится сигнал *Истина*, при этом блокируются кнопки *Пуск* и *Стоп*;
- *Отключен электрической защитой* - фиксируется в момент когда на вход *Откл эл. защитой* приходит сигнал *Истина*, сбрасывается только кнопкой *Сброс ошибок* либо когда вход *Квитирование Аварии= Истина*; При квитировании происходит выдача импульса на выход *Команда Отключить*, для сброса прибора обеспечивающего электрическую защиту;
- *Отключен аварийной кнопкой* - фиксируется в момент когда на вход *Откл эл. защитой* приходит сигнал *Истина*, сбрасывается только кнопкой *Сброс ошибок* либо когда вход *Квитирование Аварии= Истина*;
- *Ток за пределами* - формируется при появления сигнала *Истина* на входе *Ток за пределами*;
- *ТЗ Включить и ТЗ Отключить одновременно* - информирует об одновременного поступления сигнала *Истина* на входы *ТЗОтключить* и *ТЗВключить*, имеющих одинаковый приоритет и взаимно блокирующих друг друга;
- *Несанкционированное отключение вентилятора* - формируется при появлении сигналов на входах *Отключен= Истина* и *Включен=Пожь*, если при этом отсутствовали команд на отключение от оператора, ФГУ или технологических защит;
- *Несанкционированное включение вентилятора* - формируется при появлении сигналов на входах *Отключен=Пожь* и *Включен= Истина*, если при этом отсутствовали команд на включение от оператора, ФГУ или технологических защит;
- *Сработало реле Включен* - вход *Включен = Истина*;
- *Сработало реле Отключен* - вход *Отключен = Истина*;
- *Блокирована команда Включить* - вход *Запрет Включить = Истина*;
- *Блокирована команда Отключить* - вход *Запрет Отключить = Истина*;
- *Запрет управления выставлен оператором* - в окне управления нажата кнопка *Запретить управление*.

Страница свойств *Настройки* ФБ *Вентилятор односкоростной* выглядит следующим образом:



Функции элементов управления *Группы Общие*:

- *Длительность управляющего импульса (с)* – задает длительность вырабатываемого импульса;
- *Использовать блокировку включения* – использовать/не использовать вход *Запрет Включить*;
- *Использовать блокировку отключения* – использовать/не использовать вход *Запрет Отключить*;
- *Контролировать ток* – включить контроль и формирование аварий по значению эл. тока;
- *Контрольное время включения/ отключения (с)* – при не переключении ИМ по истечению данного времени формируется сигнал *Авария*;
- *Отключен аварийной кнопкой* – формировать аварийное состояние при отключении аварийной кнопкой;
- *Отключен электрической защитой* – формировать аварийное состояние при отключении аварийной защитой.

Вентилятор двухскоростной

Функциональный блок *Вентилятор двухскоростной*, загружаемый в контроллер, осуществляет функцию управления двухскоростным исполнительным механизмом, а также предоставляет возможность отображения на видеокадре мнемонического символа состояния исполнительного механизма, привязанного к этому ФБ.

При поступлении команды на вход *Включить* происходит включение первой скорости, если механизм был выключен и переключение с первой на вторую скорость или со второй на первую (по очереди), если механизм был включен. Поступление команды на вход *Отключить* приводит к отключению механизма (и первой и второй скорости). В процессе управления двухскоростным вентилятором возникает ситуация переключения с первой на вторую скорость и со второй на первую. Переключение реализуется последовательностью: отключение предыдущей скорости – включение следующей. При такой последовательности возникает фаза отключенного состояния механизма. Для скрытия состояния *Отключено* в момент переключения скоростей введено понятие *Идет переключение*.

Переключения осуществляются в соответствии со следующей логикой. При поступлении команды на вход *Включить* проверяется наличие признака *Идет переключение* и, при его отсутствии, этот признак выставляется. При наличии признака *Идет переключение* команда *Включить* игнорируется. Проверяется состояние входов *Включена 1ск.* и *Включена 2ск.*:

- Если не включена ни одна скорость, то выдаётся команда *Включить 1ск.* и, по истечении времени *Td* (Контрольное время включения/отключения), проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Неисполнение команды включения 1(2) скорости* и сбрасывается признак *Идет переключение* (мигание зеленого кружка внутри мнемосимвола). При успешном выполнении команды сбрасывается признак *Идет переключение*.
- Если включена первая скорость, тогда проверяется наличие сигнала на входе *Перегрузка 1ск.* Если на этом входе есть сигнал, то сбрасывается признак *Идет переключение*, а команда *Включить* игнорируется. При отсутствии перегрузки 1 скорости

и наличии включённого состояния 1 скорости выдаётся команда *Отключить 1ск.*, выставляется признак *Идет переключение* и, по истечении времени T_d , проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Неисполнение команды отключения 1 скорости* и сбрасывается признак *Идет переключение*. При успешном выполнении команды, выдаётся команда *Включить 2ск.* и, по истечении времени T_d , проверяется исполнение команды. При успешном выполнении команды сбрасывается признак *Идет переключение*. При обнаружении неисполнения команды выставляется признак *Неисполнение команды включения 2 скорости* и выдаётся команда на включение 1 скорости. По истечении времени T_d проверяется исполнение команды. При успешном выполнении команды сбрасывается признак *Идет переключение*. При обнаружении неисполнения команды выставляется признак *Неисполнение команды включения 1 скорости* и сбрасывается признак *Идет переключение*.

При включении механизма из *холодного* состояние переключение с 1 скорости на 2 скорость может производиться не ранее, чем через время T_{d2} (Время запрета второй скорости из холодного состояния). Если включена вторая скорость, то выдаётся команда *Отключить 2ск.* и, по истечении времени T_d , проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Неисполнение команды отключения 2 скорости* и сбрасывается признак *Идет переключение*. При успешном выполнении команды, выдаётся команда *Включить 1ск.* и, по истечении времени T_d , проверяется исполнение команды. При успешном выполнении команды сбрасывается признак *Идет переключение*. При обнаружении неисполнения команды выставляется признак *Неисполнение команды включения 1 скорости* и выдаётся команда на включение 2 скорости. По истечении времени T_d , проверяется исполнение команды. При успешном выполнении команды сбрасывается признак *Идет переключение*. При обнаружении неисполнения команды выставляется признак *Неисполнение команды включения 2 скорости* и сбрасывается признак *Идет переключение*.

Поступление команды на вход *Отключить* приводит к отключению механизма не зависимо от скорости на которой он работал. Действия, выполняемые по команде от входа *Включить* отменяются. Проверяется состояние входов *Включена 1ск.* и *Включена 2ск.*. При наличии включённого состояния какой-либо скорости выдаётся команда на их отключение и, по истечении времени T_d , проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Неисполнение команды отключения 1(2) скорости*. При подтверждении выполнения команды *Отключить* флаг *Идет переключение* снимается.

Далее происходит выставление признака *Запрет управления* через окно оператора, блокирует выдачу каких-либо команд из функционального блока. Эта команда обладает наивысшим приоритетом. После чего выполняется Диагностика, диагностируются:

- Исполнение выданных команд на включение/отключение 1/2 скорости;
- Наличие сигналов на входах *Отключён электрической защитой 2/1ск.*, *Перегрузка 1ск.*;
- Несанкционированный доступ – в случае изменения состояния *включена/отключена 1/2 скорость* без команд от функционального блока и при отсутствии команд от электрических защит.

Появление следующих диагностических сигналов блокирует команды по входам включения/отключения 1/2 скорости до квитирования - Неисполнение команды включения/отключения 1/2 скорости, Отключен электрической защитой 2/1ск.

После диагностики выходные команды формируются в виде импульса заданной длительности (в настройках ФБ). Одновременная выдача противоположных команд блокируется.

Возможности управления:

- Автоматическое, через управляющие входы;
- Ручное, через окно операторского управления.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - подерживается.

Входы ФБ:

Имя	Тип	Примечание
Включить	Логический	Происходит включение первой скорости, если механизм был выключен, и переключение с первой на вторую скорость и со второй на первую (по очереди), если механизм был включен.
Запрет Вклю-	Логический	Блокировка включения или переключения между скоростями.

Имя	Тип	Примечание
читать		
Отключить	Логический	Приводит к отключению механизма (и первой, и второй скорости). Действия, выполняемые по команде от входа <i>Включить</i> , отменяются.
ТЗВключить 1ск	Логический	Команда от ТЗ, управляющая включением 1 скорости. Данная команда имеет повышенный приоритет. Работа этой команды не зависит от запретов по входам управления. Блокируется только оператором через окно управления.
ТЗОтключить 1ск	Логический	Команда от ТЗ, управляющая отключением 1 скорости. Данная команда имеет повышенный приоритет. Работа этой команды не зависит от запретов по входам управления. Блокируется только оператором через окно управления.
ТЗВключить 2ск	Логический	Команда от ТЗ, управляющая включением 2 скорости. Данная команда имеет повышенный приоритет. Работа этой команды не зависит от запретов по входам управления. Блокируется только оператором через окно управления.
ТЗОтключить 2ск	Логический	Команда от ТЗ, управляющая отключением 2 скорости. Данная команда имеет повышенный приоритет. Работа этой команды не зависит от запретов по входам управления. Блокируется только оператором через окно управления.
Включена 1ск	Логический	Флаг включения 1 скорости.
Отключена 1ск	Логический	Флаг отключения 1 скорости.
Эл защита 1ск	Логический	Факт срабатывания Эл. Защиты двигателя на 1 скорости.
Перегрузка 1ск	Логический	Если включена первая скорость, тогда проверяется наличие сигнала на входе <i>Перегрузка 1ск</i> . Если на этом входе есть сигнал, то сбрасывается признак <i>Идет переключение</i> , а команда <i>Включить</i> игнорируется.
Включена 2ск	Логический	Флаг включения 2 скорости.
Отключена 2ск	Логический	Флаг отключения 2 скорости.
Эл защита 2ск	Логический	Факт срабатывания Эл. Защиты двигателя на 2 скорости.
Отключение ав. кнопкой	Логический	Сигнал отключения, приходящий от пульта управления ИМ.
Авария	Логический	Внешний аварийный сигнал (резервный), используемый разработчиком.
Ток 1ск за пределами	Логический	Выдается в виде индикации соответствующем символом в окне объекта.
Ток 2ск за пределами	Логический	Выдается в виде индикации соответствующем символом в окне объекта.
Квитирование Аварии	Логический	Внешняя команда квитирования ошибок.
Выходы ФБ:		
Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-хбайтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма: 1 – включен; 2 – отключен; 8 – недостоверное состояние; 9 – включен на второй скорости. Второй байт представляет состояние режима механизма и для данного ФБ не используется. Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000010 – авария (от внешнего входа, не включился за контрольное время); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте -262144) авария (второй бит в третьем байте - 131072) при включенном состоянии вентилятора (262144+131072+1). На выходе будет 393217.
Выход Авария	Логический	Обобщенный сигнал <i>Авария</i> о возникших ошибках в работе ИМ.
Запрет управления	Логический	Сигнализирует о блокировке выдачи каких-либо команд из функционального блока. Данный флаг выставляется в окне управления оператора и обладает наивысшим приоритетом.
Команда Включить 1ск	Логический	Команда, выдаваемая на исполнение во внешние исполнительные цепи.
Команда Отключить 1ск	Логический	Команда, выдаваемая на исполнение во внешние исполнительные цепи.
Команда Включить	Логический	Команда, выдаваемая на исполнение во внешние исполнительные

2ск		цепи.
Команда Отключить	Логический	Команда, выдаваемая на исполнение во внешние исполнительные цепи.
2ск		
Остановлен	Логический	Сигнал, показывающий отсутствие всех трех состояний включенного ИМ (1 скорость, 2 скорость, состояние перехода между скоростями).
Включен	Логический	Обобщенный сигнал, показывающий наличие хотя бы одного из состояний включенного ИМ (1 скорость, 2 скорость, состояние перехода между скоростями).

Мнемоническое отображение ФБ на мнемосхеме:

Состояние	Поведение
Вентилятор Включен 1 скорость	Ровный зеленый цвет с цифрой скорости
Вентилятор Включен 2 скорость	Ровный зеленый цвет с цифрой скорости
Вентилятор Отключен	Ровный белый цвет
Присутствует сигнал аварии	Мигает красная рамочка.
Недостовверное состояние	Ровный серый цвет
Включен 1 скорость и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Включен 2 скорость и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Отключен и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Недостовверное состояние и запрет управления	Запрещены все операции, включая управление от защит и блокировок.

Элемент динамический, при клике вызывается окно управления.

По кнопке *Диагностика* вызывается панель состояния диагностики выбранного механизма. Панель диагностики содержит индикаторы:

- *Аппаратная недостоверность 1ск* - входные сигналы, отображающие состояние электродвигателя для 1 скорости, имеют плохой признак качества;
- *1ск не включилась за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Включена 1ск* по истечении контрольного времени;
- *1ск не отключилась за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Отключена 1ск* по истечении контрольного времени;
- *1ск сработала электрическая защита* - в случае появления сигнала *Истина* на входе *Эл защита 1ск*;
- *1ск ток за пределами* - в случае появления сигнала *Истина* на входе *Ток 1ск за пределами*;
- *Аппаратная недостоверность 2ск* - входные сигналы, отображающие состояние электродвигателя для 2 скорости, имеют плохой признак качества;
- *2ск не включилась за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Включена 2ск* по истечении контрольного времени;
- *2ск не отключилась за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Отключена 2ск* по истечении контрольного времени;
- *2ск сработала электрическая защита* - в случае появления сигнала *Истина* на входе *Эл защита 2ск*;
- *2ск ток за пределами* - в случае появления сигнала *Истина* на входе *Ток 2ск за пределами*;
- *Отключен аварийной кнопкой* – вход *Откл ав кнопкой* = *Истина*;
- *Авария от внешнего входа* - вход *Авария* = *Истина*;
- *1ск ТЗ Включить и ТЗ Отключить одновременно* - в случае одновременного поступления сигнала *Истина* на входы *ТЗОтключить 1ск* и *ТЗВключить 1ск*, имеющих одинаковый приоритет;
- *2ск ТЗ Включить и ТЗ Отключить одновременно* - в случае одновременного поступления сигнала *Истина* на входы *ТЗОтключить 2ск* и *ТЗВключить 2ск*, имеющих одинаковый приоритет;
- *Несанкционированное отключение 1ск* - появление сигнала *Истина* на входе *Отключена 1ск*, при отсутствии команд на отключение 1 скорости от оператора, ФГУ или технологических защит;
- *Несанкционированное включение 1ск* - появление сигнала *Истина* на входе *Включена 1ск*, при отсутствии команд на включение 1 скорости от оператора, ФГУ или технологических защит;
- *Несанкционированное включение 2ск* - появление сигнала *Истина* на входе *Включена 2ск*, при отсутствии команд на включение 2 скорости от оператора, ФГУ или технологических защит;

- *Несанкционированное включение 2ск* - появление сигнала *Истина* на входе *Включена 2ск*, при отсутствии команд на включение 2 скорости от оператора, ФГУ или технологических защит;
- *Сработало реле Включена 1ск* - вход *Включена 1ск* = *Истина*;
- *Сработало реле Отключена 1ск* - вход *Отключена 1ск* = *Истина*;
- *Сработало реле Включена 2ск* - вход *Включена 2ск* = *Истина*;
- *Сработало реле Отключена 2ск* - вход *Отключена 2ск* = *Истина*;
- *Блокирована команда Включить* - вход *Запрет Включить* = *Истина*;
- *Перегрузка 1ск* - вход *Перегрузка 1ск* = *Истина*;
- *Запрещено включение 2ск* – появляется в течение времени запрета переключения на 2ск после включения из холодного состояния (задается на странице свойств *Настройки*);
- *Запрет управления выставлен оператором* - в окне управления нажата кнопка *Запретить управление*.

Страница свойств «Настройки» ФБ Вентилятор двухскоростной выглядит следующим образом:

Функции элементов управления группы *Общие*:

- *Время задержки переключения скоростей (мс)* – это время выдерживается после отключения одной скорости и перед включением другой;
- *Время запрета 2ск из холодного старта (с)* – время запрета переключения на 2ск после холодного старта;
- *Длительность управляющего импульса (мс)* – задает длительность вырабатываемого импульса;
- *Запрет переключения на 2ск после холодного старта* – запрет переключения на 2ск в течении N-секунд после холодного старта;
- *Контролировать ток* – включить контроль и формирование аварий по значению эл. тока;
- *Контрольное время включения/отключения (с)* – при не переключении ИМ по истечению данного времени формируется сигнал *Авария*;
- *Отключен аварийной кнопкой* – формировать аварийное состояние при отключении аварийной кнопкой;
- *Отключен электрической защитой* – формировать аварийное состояние при отключении аварийной защитой;

Задвижка с промостановом

Функциональный блок *Задвижка с промостановом* осуществляет функцию управления задвижкой с промежуточным остановом и без останова, контролирует выполнение команд за заданное время, формирует обобщенный сигнал состояния задвижки. ФБ также предоставляет возможность отображения на мнемосхеме состояния исполнительного механизма, привязанного к этому ФБ.

Логика работы функционального блока следующая. При поступлении команды на вход *Открыть* на выходе ФБ устанавливается положительный потенциальный сигнал, который переводит клапан в открытое состояние. При поступлении команды на вход *Закрыть* на выходе ФБ устанавливается положительный потенциальный сигнал, который переводит клапан в закрытое состояние.

Переключения осуществляются в соответствии со следующей логикой:

- При поступлении команды на вход *Открыть*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Закрыть* и *Запрет*, выдается команда на открытие ИМ (у мнемонического символа мигает зеленое поле). По истечении контрольного времени включения/отключения, проверяется исполнение команды. Если команда выполнена, мнемонический символ становится зеленым. При обнаружении неисполнения команды выставляется признак *Не открылся за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета.
- При поступлении команды на вход *Закрыть*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Открыть* и *Запрет*, выдается команда на закрытие ИМ (у мнемонического символа мигает зеленое поле). По истечении

контрольного времени включения/отключения, проверяется исполнение команды. Если команда выполнена, мнемонический символ становится белым. При обнаружении неисполнения команды выставляется признак *Не закрылся за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета.

- При поступлении команды *Стоп*, отсутствия запрета управления, выставленного оператором, отсутствия команды на открытие и закрытие, движение ИМ на открытие/закрытие останавливается.

Возможности управления:

- Автоматическое, через управляющие входы;
- Ручное, через окно операторского управления.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

Входы ФБ:

Имя	Тип	Примечание
Открыть	Логический	Команда. Переводит выход <i>Команда Открыть</i> в состояние <i>Истина</i> на выход при отсутствии сигнала <i>Запрет Открыть</i> и команд от технологических защит.
Стоп	Логический	Команда. Останавливает процесс открытия/закрытия задвижки, при условии наличия промостанова (задается на странице свойств <i>Настройки</i>).
Закрыть	Логический	Команда. Переводит выход <i>Команда Закрыть</i> в состояние <i>Истина</i> на выход при отсутствии сигнала <i>Запрет Закрыть</i> и команд от технологических защит.
Запрет Открыть	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Открыть</i> от входа и от оператора и на панели состояния диагностики загорается индикатор <i>Блокирована команда Открыть</i> . Данная блокировка не оказывает влияние на команду от ТЗ. Вход появляется, если на странице свойств <i>Настройки</i> установлена <i>Истина</i> в графе <i>Использовать блокировку открытия</i> .
Запрет Закрыть	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Закрыть</i> от входа и от оператора на панели состояния диагностики загорается индикатор <i>Блокирована команда Закрыть</i> . Данная блокировка не оказывает влияние на команду от ТЗ. Вход появляется, если на странице свойств <i>Настройки</i> установлена <i>Истина</i> в графе <i>Использовать блокировку закрытия</i> .
ТЗОткрыть	Логический	Команда <i>Открыть</i> от технологических защит. Имеет повышенный приоритет. Данная команда блокируется только от запрета управления, выставленного оператором в окне управления. В случае одновременного поступления с командой <i>ТЗЗакрыть</i> , имеющей тот же приоритет, обе команды игнорируются, а на панели состояния диагностики загорается индикатор <i>ТЗ Открыть и ТЗ Закрыть одновременно</i> .
ТЗЗакрыть	Логический	Команда <i>Закрыть</i> от технологических защит. Имеет повышенный приоритет. Данная команда блокируется только от запрета управления, выставленного оператором в окне управления. В случае одновременного поступления с командой <i>ТЗОткрыть</i> , имеющей тот же приоритет, обе команды игнорируются, а на панели состояния диагностики загорается индикатор <i>ТЗ Открыть и ТЗ Закрыть одновременно</i> .
Открыт	Логический	КВ0 (концевой выключатель открытого положения). В случае одновременного поступления с сигналом <i>Закрыт</i> , имеющей тот же приоритет, на панели состояния диагностики загорается индикатор <i>Нет питания</i> .
Закрыт	Логический	КВ3 (концевой выключатель закрытого положения). В случае одновременного поступления с сигналом <i>Открыт</i> , имеющей тот же приоритет, на панели состояния диагностики загорается индикатор <i>Нет питания</i> .
Авария	Логический	Авария от внешнего входа. Если <i>Истина</i> , на панели состояния диагностики загорается индикатор <i>Авария от внешнего входа</i> .
Квитирование Аварии	Логический	Снимает мигание сигнальной области и аварии с фиксацией (например <i>Не открылся за контрольное время</i>).

Выходы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-байтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма:

Имя	Тип	Примечание
		1 – открыт; 2 – закрыт; 3 – открывается; 4 – закрывается; 5 – остановлен в промежуточном состоянии; 6 – трогается на открытие; 7 – трогается на закрытие; 8 – недостоверное состояние. Второй байт представляет состояние режима механизма и для данного ФБ не используется. Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000010 – авария (от внешнего входа, не открылся за контрольное время); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте - 262144) авария (второй бит в третьем байте - 131072) при открытом состоянии клапана (262144+131072+1). На выходе будет 393217.
Выход Авария	Логический	Принимает значение <i>Истина</i> при наличии какой-либо аварии.
Запрет управления	Логический	Принимает значение <i>Истина</i> , если в окне управления нажата кнопка <i>Запретить управление</i> , либо произошла <i>Авария</i> . При этом все команды, подаваемые на входы ФБ, игнорируются.
Команда Открыть	Логический	Потенциальная команда на открытие, выдаваемая на исполнительные цепи.
Команда Закрыть	Логический	Потенциальная команда на закрытие, выдаваемая на исполнительные цепи.

Мнемоническое отображение ФБ на мнемосхеме. Символ динамического элемента электрифицированной арматуры состоит из двух независимых динамических элементов в виде треугольников с соединенными вершинами. При горизонтальном расположении на трубопроводе левый сегмент символизирует закрытое состояние, а правый – открытое. При вертикальном расположении символа арматуры нижний сегмент символизирует закрытое состояние, верхний – открытое.

Состояние	Поведение
Закрыто	Ровный белый цвет.
Закрыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Открыто	Ровный зеленый цвет.
Открыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Промежуточное положение	При движении из положения <i>Закрыто</i> в <i>Открыто</i> мигает зеленое поле, из <i>Открыто</i> в <i>Закрыто</i> мигает белое поле, в статическом положении мигания нет.
Присутствует сигнал аварии	Мигает красная рамочка.
Недостоверное состояние	Ровный серый цвет.
Ручное управление	Статический элемент, отображается цветом среды.

Элемент динамический, при клике вызывается окно управления. Для арматуры без промостанова кнопка *СТОП отсутствует*. Наличие промостанова указывается в настройках.

По кнопке *Диагностика* вызывается панель состояния диагностики задвижки.

- *Аппаратная недостоверность* – входные сигналы, отображающие состояние электродвигателя, имеют плохой признак качества;
- *Не открылся за контрольное время* – в случае отсутствия сигнала *Истина* на входе *Открыт* по истечении времени хода исполнительного механизма;
- *Не закрылся за контрольное время* – в случае отсутствия сигнала *Истина* на входе *Закрыт* по истечении времени хода исполнительного механизма;
- *Не снялся с концевика открыт за контрольное время* – вход *Открыт* = *Истина* по истечении контрольного времени при наличии управляющей команды на закрытие;
- *Не снялся с концевика закрыт за контрольное время* – вход *Закрыт* = *Истина* по истечении контрольного времени при наличии управляющей команды на открытие;
- *Нет питания* – на вход ФБ одновременно пришли два взаимоисключающих сигнала *Открыт* = *Истина* и *Закрыт* = *Истина*;
- *Авария от внешнего входа* – вход *Авария* = *Истина*;

- *ТЗ Открыть и ТЗ Закрыть одновременно* – в случае одновременного поступления с команд *ТЗОтключить = Истина* и *ТЗВключить = Истина*, имеющих одинаковый приоритет;
- *Сработал концевик Открыт* – вход *Открыт = Истина*;
- *Сработал концевик Закрыт* – вход *Закрыт = Истина*;
- *Блокирована команда Открыть* – вход *Запрет Открыть = Истина*;
- *Блокирована команда Закрыть* – вход *Запрет Закрыть = Истина*;
- *Запрет управления выставлен оператором* – в окне управления нажата кнопка *Запретить управление*.

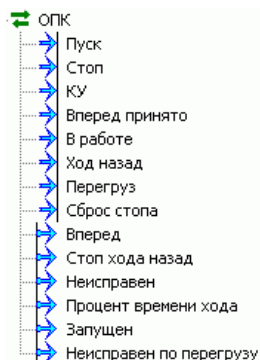
Страница свойств *Настройки* ФБ *Задвижка с промостановом* выглядит следующим образом:

Функции элементов управления группы *Общие*:

- *Время реверса (с)* – выдержка времени при смене направления движения на противоположное, блокирует управление;
- *Время хода исполнительного механизма (с)* – время хода ИМ влияет на оценочный показатель степени открытия и аварии;
- *Использовать блокировку закрыть* – использовать/ не использовать вход *Запрет Закрыть*;
- *Использовать блокировку открытия (мс)* – использовать/ не использовать вход *Запрет Открыть*;
- *Контрольное время снятия с концевика (с)* – время, через которое ФБ выдаст сигнал *Авария*, если ИМ не снимется с концевика при наличии управляющей команды;
- *Люфт ИМ (С)* – настроечный параметр, влияющий на расчет оценочного положения ИМ;
- *Наличие промостанова* – исходная настройка, блокирующая функцию промостанова.

Обдувка паровая котла

Этот ФБ предназначен для управления механизмом подачи пара в котел. Вид функционального блока в дереве объектов:



В каждый момент времени ФБ может находиться в одном из состояний:

- Покой;
- Вперед;
- Назад до черты;
- Назад после черты;
- Стоп назад с памятью;
- Стоп назад без памяти.

Состояния *Неисправен* и *Неисправен по перегрузу* могут быть одновременно с любым из вышеперечисленных состояний, кроме состояния *Вперед*.

Описание состояний:

Состояние	Значения выходов
Покой	Вперед = Ложь Стоп хода назад = Ложь Процент времени хода = 0

Вперед	Запущен = Ложь
	Вперед = Истина
	Стоп хода назад = Ложь
	Процент = увеличивается на (100/Время хода вперед) Запущен = Истина
Назад до черты	Вперед = Ложь
	Стоп хода назад = Ложь
	Процент = уменьшается на (100/Время хода назад для индикатора)
	Запущен = Истина
Назад после черты	Вперед = Ложь
	Стоп хода назад = Ложь
	Процент = уменьшается на (100/Время хода назад для индикатора)
	Запущен = Ложь
	Выход Запущен отключается по истечении Время хода назад для запуска (задается на странице свойств настройки), и по его заднему фронту можно отследить момент, когда запускать следующий.
Стоп назад с памятью	Вперед = Ложь
	Стоп хода назад = Истина
	Процент = запоминается последнее положение
	Запущен = Ложь
Стоп назад без памяти	Отличается от аналогичного состояния с памятью тем, что выход «Процент времени хода» не запоминается, а равен 0.
Неисправен	Состояние появляется при наступлении следующих событий:
	Сработала перегрузка
	Не пришло Вперед
	Пропало Вперед
	Не пришло в работе
	Прпало в работе
Нет хода назад	

Ход назад делится на две части: до- и после условной черты запуска следующего механизма. После разворота ИМ сначала идет до черты, а после того как выход *Процент времени хода* пересечет величину $[100\% - ((\text{Время хода назад для запуска} / \text{Время хода назад для индикатора}) * 100\%)]$ -после черты.

Условия возникновения событий:

Событие	Условие
Сработала перегрузка	Вход <i>Перегруз</i> = <i>Истина</i> , и текущее состояние <i>Вперед</i> или <i>Назад до черты</i> или <i>Назад после черты</i> . Событие <i>Сработала перегрузка</i> не наступает в течение времени, равному <i>Контрольному времени перегрузки</i> , если механизм переходит из следующих в состояний: из <i>Покой</i> в <i>Вперед</i> ; из <i>Вперед</i> в <i>Назад до черты</i> ; из <i>Вперед</i> в <i>Назад после черты</i> ; из <i>Стоп назад с памятью/без памяти</i> в <i>Назад до черты/после черты</i> .
Не пришло Вперед	Наступает, если текущее состояние <i>Вперед</i> , и с момента старта прошло <i>Контрольное время пуска</i> , и вход <i>Вперед принято</i> равен <i>Ложь</i> .
Пропало Вперед	Наступает, если текущее состояние <i>Вперед</i> , и с момента старта прошло больше времени, чем <i>Контрольное время пуска</i> , и вход <i>Вперед принято</i> равен <i>Ложь</i> .
Не пришло в работе	Наступает, если текущее состояние <i>Вперед</i> или <i>Назад до черты</i> , или <i>Назад после черты</i> и с момента старта прошло <i>Контрольное время пуска</i> , и вход <i>В работе</i> равен <i>Ложь</i> .
Пропало в работе	Наступает, если текущее состояние <i>Вперед</i> , <i>Назад до черты</i> , <i>Назад после черты</i> или <i>Стоп назад с памятью</i> , и с момента старта прошло больше времени, чем <i>Контрольное время пуска</i> , и вход <i>В работе</i> равен <i>Ложь</i> .
Нет хода назад	Наступает, если текущее состояние <i>Назад до черты</i> или <i>Назад после черты</i> , и с момента разворота или начала движения назад из состояния <i>Стоп назад с памятью</i> прошло больше времени, чем <i>Контрольное время пуска</i> , и вход <i>Ход назад</i> равен <i>Ложь</i> .
Истекло время хода назад	Происходит, когда по расчету ИМ должен вернуться в первоначальное положение. Если не было аварийных разворотов, то это будет спустя <i>Время хода назад для индикатора</i> после разворота. Если разворот был аварийный, то будет рассчитываться в зависимости от того, в каком месте ИМ развернулся.
До черты пуска (при ходе Вперед)	<i>Истина</i> , если текущее состояние <i>Вперед</i> и с момента пуска прошло времени меньше, чем <i>Время хода вперед</i> минус <i>Время хода назад для запуска</i> (то есть, не достигнута условная черта, на которой на обратном ходе пускают следующий механизм).

Условия перехода из одного состояния в другое:

Выход из состояния	Событие	Приход в состояние
Покой	$KU = \text{Истина}$; $\text{Стоп} = \text{Истина}$; Передний фронт входа <i>Пуск</i> .	Вперед.
Вперед	Произошло событие <i>Сработала перегрузка</i> и механизм находился в положении <i>До черты пуска</i> .	Назад после черты.
	Сработала перегрузка и механизм находился в положении <i>После черты пуска</i> .	Назад до черты.
	Наступило одно из событий: <i>Пропало Вперед</i> или	Назад после черты.

	<i>Пропало в работе</i> , и механизм находился в состоянии <i>До черты пуска</i> .	
	Наступило одно из событий: <i>Пропало Вперед</i> или <i>Пропало в работе</i> , и механизм находился в состоянии <i>После черты пуска</i> .	Назад до черты.
	<i>Стоп = Истина</i> .	Стоп назад с памятью.
	<i>Не пришло Вперед</i> или <i>Не пришло в работе</i> .	Покой.
	Истекло <i>Время хода вперед</i> .	Назад до черты.
Назад до черты	Сработала перегрузка.	Стоп назад с памятью.
	Вход <i>Стоп = Истина</i> .	Стоп назад с памятью.
	Истекло <i>Время хода назад для запуска</i> .	Назад после черты.
	Пропало в работе.	Покой.
Назад после черты	Сработала перегрузка.	Стоп назад с памятью.
	<i>Стоп = Истина</i> .	Стоп назад с памятью.
	Истекло <i>Время хода назад</i> .	Покой.
	Пропало в работе.	Покой.
Стоп назад с памятью	Передний фронт входа <i>Сброс стопа назад</i> .	<i>Назад после черты</i> (Вход <i>Запущен</i> при стопе снимаем и больше не включаем).
	Пропало в работе.	Стоп назад без памяти.
Стоп назад без памяти	Передний фронт входа <i>Сброс стопа назад</i> .	Покой.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Входы ФБ:

Обозначение	Тип	Описание
Пуск	Логический	Импульс, который запускает работу механизма ОПК. ИМ начинает движение вперед. Сигнал на данный вход нужно формировать в дереве объектов MasterSCADA.
Стоп	Логический	Останов движения исполнительного механизма.
КУ	Логический	Если вход равен <i>Истина</i> , то есть возможность запустить механизм в работу. Если вход равен <i>Ложь</i> , то считается, что механизм в работу не введен оператором. Если механизм уже выполняет алгоритм работы, то состояние данного входа значения не имеет.
Вперед принято	Логический	После того как исполнительный механизм начинает движение из дока и проходит концевик формируется сигнал <i>Истина</i> . К входу необходимо подсоединить соответствующий сигнал, поступающий с выхода модуля ввода/вывода контроллера.
В работе	Логический	Если вход равен истина, то электрическое питание к двигателям ИМ подведено. К входу необходимо подсоединить соответствующий сигнал, поступающий с выхода модуля ввода/вывода контроллера.
Ход назад	Логический	Сигнализирует о движении ИМ назад в док. К входу необходимо подсоединить соответствующий сигнал, поступающий с выхода модуля ввода/вывода контроллера.
Перегруз	Логический	К входу необходимо подсоединить сигнал, который возникает в контроллере при перегрузке электродвигателей.
Сброс стопа	Логический	Если вход равен <i>Истина</i> , то механизм начинает движение.

Выходы ФБ:

Обозначение	Тип	Описание
Вперед	Логический	Выход сигнализирует о том, что механизм двигается вперед.
Стоп хода назад	Логический	Механизм остановился при движении в обратном направлении.
Неисправен	Логический	Сигнализирует о неисправности механизма. Выход равен <i>Истина</i> , если произошло одно из следующих событий: Сработала перегрузка; Не пришло Вперед; Пропало Вперед; Не пришло в работе; Пропало в работе; Нет хода назад.
Процент времени хода	Вещественный дв. точности	Показывает состояние механизма. 0% соответствует началу пути из дока. 100% - соответствуют окончанию пути ИМ.
Запущен	Логический	ИМ находится в движении.
Неисправен по перегрузу	Логический	Выход равен <i>Истина</i> , если ФБ рассчитал событие <i>Сработала перегрузка</i> .

Страница свойств *Настройки* ФБ ОПК выглядит следующим образом:

Общие	
Время хода вперед	120
Время хода назад для запуска	10
Время хода назад для индикатора	60
Контрольное время перегруза	10
Контрольное время трогания	10

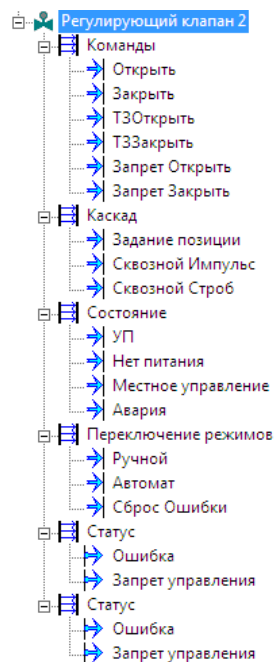
Элемент	Описание
Время хода вперед	Время хода механизма вперед.
Время хода назад для запуска	Время хода назад для запуска следующего механизма.
Время хода назад для индикатора	Время хода назад для расчета процента для индикации хода механизма.
Контрольное время перегруза	Время, в течение которого ФБ может игнорировать перегрузку.
Контрольное время трогания	Время, спустя которое ИМ должен начать движение.

Предохранительный запорный клапан

Визуальный функциональный блок *Предохранительный запорный клапан* осуществляет управление клапаном данного типа. ФБ также предоставляет возможность отображения на мнемосхеме состояния исполнительного механизма, привязанного к этому ФБ.

При поступлении команды на вход *Закреть* на выходе ФБ устанавливается положительный потенциальный сигнал, который переводит клапан в закрытое состояние. При поступлении команды на вход *Закреть*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Запрет Закреть*, выдётся команда на закрытие ИМ (мнемонический символ окрашивается в белый цвет). По истечении контрольного времени открытия/закрытия, проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Не закрылся за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета.

Вид функционального блока в дереве объекта:



Возможности управления:

- Автоматическое, через управляющие входы;
- Ручное, через окно операторского управления.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Входы ФБ:

Имя	Тип	Примечание
Закреть	Логический	Команда. Переводит выход <i>Команда Закреть</i> в состояние <i>Истина</i> при отсутствии сигнала <i>Запрет Закреть</i> и команд от технологических защит.

Запрет Закрыть	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Закрыть</i> от входа и от оператора и на панели состояния диагностики загорается индикатор <i>Блокирована команда Закрыть</i> . Данная блокировка не оказывает влияние на команду от ТЗ. Вход появляется, если на странице свойств <i>Настройки</i> установлена <i>Истина</i> в графе <i>Использовать блокировку закрытия</i> .
ТЗЗакрыть	Логический	Команда <i>Закрыть</i> от технологических защит. Имеет повышенный приоритет. Данная команда блокируется только от запрета управления, выставленного оператором в окне управления.
Открыт	Логический	КВО (концевой выключатель открытого положения).
Закрыт	Логический	КВЗ (концевой выключатель закрытого положения).
Авария	Логический	Авария от внешнего входа. Если <i>Истина</i> , на панели состояния диагностики загорается индикатор <i>Авария от внешнего входа</i> .
Неисправность цепи эл магнита	Логический	Если <i>Истина</i> , соответствующий индикатор загорается на панели состояния диагностики.
Цепь электромагнита исправна	Логический	Если <i>Истина</i> , соответствующий индикатор загорается на панели состояния диагностики.
Квитирование аварии	Логический	Снимает мигание сигнальной области и аварии с фиксацией (например <i>Не закрылся за контрольное время</i>).

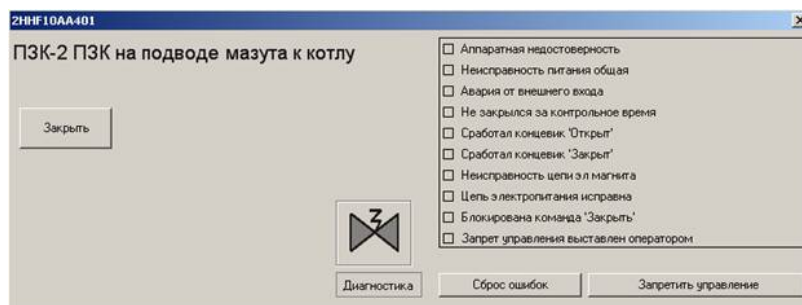
Выходы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-байтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма: 1 – открыт; 2 – закрыт; 8 – недостоверное состояние. Второй байт представляет состояние режима механизма и для данного ФБ не используется. Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000010 – авария (от внешнего входа, общая неисправность питания); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, выставлен запрет управления (первый бит в третьем байте - 65536) при открытом состоянии клапана (65536+1). На выходе будет 65537.
Выход Авария	Логический	Принимает значение <i>Истина</i> при наличии какой-либо аварии.
Запрет управления	Логический	Принимает значение <i>Истина</i> , если в окне управления нажата кнопка <i>Запретить управление</i> . При этом все команды, подаваемые на входы ФБ, игнорируются.
Команда Закрыть	Логический	Потенциальная команда, выдаваемая на исполнительные цепи.

Мнемоническое отображение ФБ на мнемосхеме. Символ динамического элемента электромагнитного клапана состоит из двух независимых динамических элементов в виде треугольников с соединенными вершинами и изображением в рассечке пружины.

Состояние	Состояние
Закрыто	Ровный белый цвет
Открыто	Ровный зеленый цвет
Недостоверное состояние	Ровный серый цвет
Присутствует сигнал аварии	Мигает красная рамочка
Закрыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок
Открыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок
Недостоверное состояние и запрет управления	Запрещены все операции, включая управление от защит и блокировок

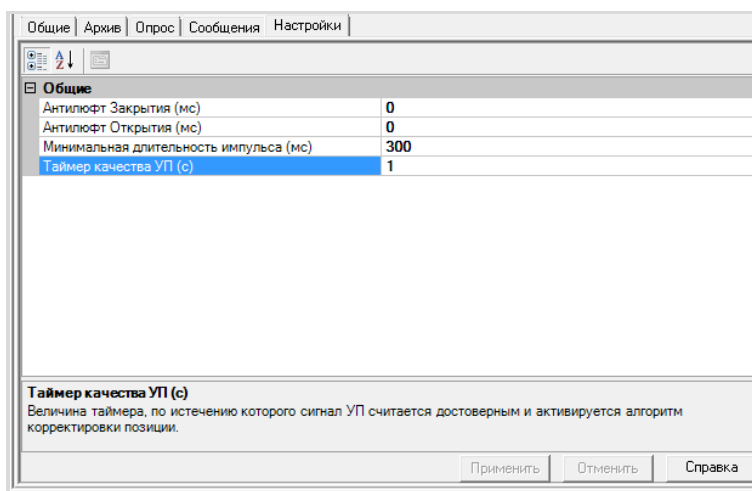
Элемент динамический. При клике вызывается окно управления



По кнопке *Диагностика* вызывается панель состояния диагностики выбранного механизма:

- *Аппаратная недостоверность* – входные сигналы, отображающие состояние клапана, имеют плохой признак качества;
- *Неисправность питания общая* – либо вход *Неисправность цепи эл магнита = Истина*, либо вход *Цепь электромагнита исправна = Ложь*;
- *Авария от внешнего входа* – вход *Авария = И Истина*;
- *Не закрылся за контрольное время* – в случае отсутствия сигнала *Истина* на входе *Закрыт* по истечении контрольного времени;
- *Сработал концевик Открыт* - вход *Открыт = Истина*;
- *Сработал концевик Закрыт* - вход *Закрыт = Истина*;
- *Неисправность цепи электромагнита* - вход *Неисправность цепи эл магнита = Истина*;
- *Цепь электропитания исправна* - вход *Цепь электромагнита исправна = Истина*;
- *Блокирована команда Закрыть* - вход *Запрет Закрыть = Истина*;
- *Запрет управления выставлен оператором* - в окне управления нажата кнопка *Запретить управление*.

Страница свойств *Настройки* ФБ *Регулирующий клапан 2* выглядит следующим образом:



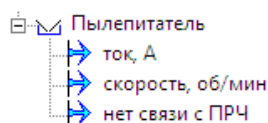
Функции элементов управления группы *Общие*:

- *Антилюфт Закрытия, (мс)*– величина компенсации люфта при смене направления движения из *Закрывается* в *Открывается*;
- *Антилюфт Открытия, (мс)* –величина компенсации люфта при смене направления движения из *Открывается* в *Закрывается*;
- *Минимальная длительность импульса, (мс)* – минимальная длительность импульса, которую может почувствовать клапан;
- *Таймер качества УП, (с)* - величина таймера, по истечению которого сигнал УП считается достоверным и активируется алгоритм корректировки позиции.

Пылепитатель

Отображает фактические характеристики преобразователя частоты питания э/д пылепитателя: ток и число оборотов, полученные непосредственно от самого преобразователя через COM-порт по интерфейсу RS-485.

Вид функционального блока в дереве объектов:



Выходы:

Обозначение	Тип	Описание
Ток, (А)	Вещественный дв. точности	Возвращает значение тока э/д привода пылепитателя.
Скорость, (об/мин)	Целый	Возвращает число оборотов э/д привода пылепитателя.
Нет связи с ПРЧ	Логический	Значение <i>Вкл</i> означает, что длительное время отсутствует ответ на запросы контроллера.

Страница свойств *Настройки* ФБ *Пылелепитатель* выглядит следующим образом:

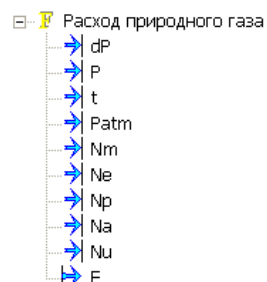
Общие Опрос Настройки	
Общие	
адрес на шине	1
время ожидания ответа в ПРЧ, мс (0-150 через 10)	10
номер СОМ-порта	1
номинальная скорость, об/мин	930
период опроса, мс	100
период чтения, мс	2000
таймаут приёма ответа, мс	50

Элементы управления:

Элемент	Описание
Адрес на шине	Адрес пылепитателя на шине данных.
Время ожидания ответа в ПРЧ, мс (0-150 через 10)	Время обработки запроса в ПРЧ (название настройки соответствует указанному в инструкции на ПРЧ).
Номер СОМ-порта	Номер СОМ-порта контроллера, через который осуществляется связь с ПРЧ.
Номинальная скорость, об/мин	Скорость вращения э/д, соответствующая частоте на выходе ПРЧ, равной 50 Гц
Период опроса, мс	Период опроса ФБ
Период чтения, мс	Период чтения частоты и тока из ПРЧ.
Таймаут приема ответа, мс	Максимальное время ожидания ответа из ПРЧ, после которого операция чтения отменяется (для ускорения обмена при наличии неотвечающих ПРЧ на шине).

Расход природного газа

Вид ФБ в дереве Объекта:

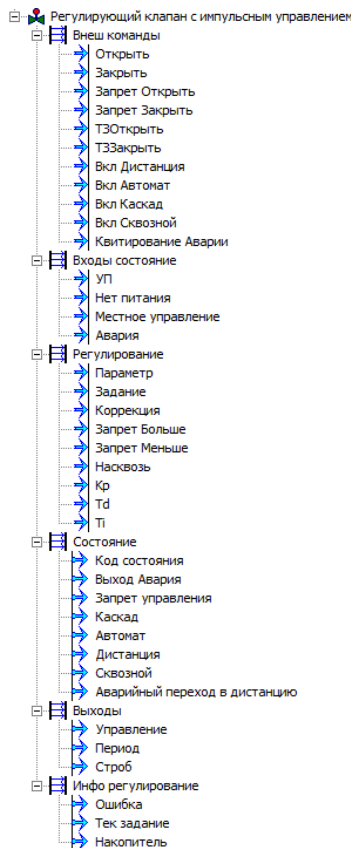


Страница свойств *Настройки* ФБ *Расход природного газа* выглядит следующим образом:

Общие Опрос Настройки	
Уставки	
Диаметр отверстия	0
Диаметр трубопровода	0
Коэффициент притупления	0
Коэффициент СУ	0
Коэффициент трубопровода	0
Эквивалентная шероховатость	0
Диаметр отверстия Диаметр отверстия сужающего устройства при 20°С мм	
<input type="button" value="Применить"/> <input type="button" value="Отменить"/> <input type="button" value="Справка"/>	

Регулирующий клапан с импульсным управлением

Вид функционального блока в дереве объекта:



Функциональный блок *Регулирующий клапан с импульсным управлением* осуществляет автоматическое управление клапаном.

ФБ предоставляет возможность работы в 4-х режимах работы: *Дистанция*, *Автомат*, *Насквозь*, *Каскад*. Каждому из режимов характерны следующие свойства:

- *Дистанция* - управляющее воздействие задается оператором вручную в окне управления регулятором;
- *Автомат* - управляющее воздействие рассчитывается внутри ФБ по ПИД – закону регулирования;
- *Задание* в режиме *Автомат* устанавливается оператором вручную из окна управления;
- *Каскад* - управляющее воздействие рассчитывается внутри ФБ по ПИД – закону регулирования;
- *Задание* в режиме *Каскад* устанавливается внешней командой по входу *Задание*;
- *Насквозь* - управляющее воздействие рассчитывается внешней логикой. Управление передается на выход ФБ со входа *Насквозь*.

Передаточная функция регулятора в ФБ выглядит следующим образом:

$$W_{per}(s) = K_p * T_{um} * (s + 1/T_i + T_d * s^2).$$

При подключении к исполнительному механизму с передаточной функцией $W_{um}(s) = 1/(T_{um} * s)$, передаточная функция регулятора приобретает вид:

$$W_{per}(s) = K_p + K_p / (T_i * s) + K_p * T_d * s$$

В рамках реализации данного функционального блока внесена функция фильтрации входного параметра. Она реализована по апериодическому закону и представляет собой фильтр первого порядка. В алгоритм функционирования ФБ внесена процедура обеспечения безударного перехода из одного режима в другой (статическая балансировка по выходу). Например, при переходе их режимов *Дистанция* или *Насквозь* в режимы *Автомат* или *Каскад*.

В функциональном блоке предусмотрена возможность диагностики процесса регулирования. Аварийный сигнал формируется в случае, когда за заданное контрольное время величина рассогласования не стала меньше требуемой логикой работы алгоритма.

В алгоритме регулирования имеется возможность настройки зоны нечувствительности. Есть возможность выбора их двух типов, которые описаны соответствующими формулами:

- $y = 0$, если $|x| \leq \text{Мертвая зона рассогласования}$, $y = x$, если $|x| > \text{Мертвая зона рассогласования}$;
- $y=0$, если $|x| \leq \text{Мертвая зона рассогласования}$, $y=(|x|-\text{delta}) \cdot \text{sign}(x)$, если $|x| > \text{Мертвая зона рассогласования}$.

Возможности управления:

- Автоматическое, через управляющие входы;
- Ручное, через окно операторского управления.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

Входы ФБ:

Имя	Тип	Примечание
Открыть	Логический	Команда. Подается на выход в режиме <i>Дистанция</i> при отсутствии сигнала <i>Запрет Открыть</i> , команд от технологических защит и сигналов от оператора.
Закрыть	Логический	Команда. Подается на выход в режиме <i>Дистанция</i> при отсутствии сигнала <i>Запрет Закрыть</i> , команд от технологических защит и сигналов от оператора.
Запрет Открыть	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Открыть</i> от входа и от оператора и на панели состояния диагностики загорается индикатор <i>Блокирована команда Открыть</i> . Данная блокировка не оказывает влияние на команду от ТЗ.
Запрет Закрыть	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Закрыть</i> от входа и от оператора и на панели состояния диагностики загорается индикатор <i>Блокирована команда Закрыть</i> . Данная блокировка не оказывает влияние на команду от ТЗ.
ТЗОткрыть	Логический	Команда <i>Открыть</i> от технологических защит. Имеет повышенный приоритет. В случае одновременного поступления с командой <i>ТЗЗакрыть</i> , имеющей тот же приоритет, обе команды игнорируются, а на панели состояния диагностики загорается индикатор <i>ТЗ Открыть и ТЗ Закрыть одновременно</i> . Данная команда блокируется только от запрета управления, выставленного оператором в окне управления.
ТЗЗакрыть	Логический	Команда <i>Закрыть</i> от технологических защит. Имеет повышенный приоритет. В случае одновременного поступления с командой <i>ТЗОткрыть</i> , имеющей тот же приоритет, обе команды игнорируются, а на панели состояния диагностики загорается индикатор <i>ТЗ Открыть и ТЗ Закрыть одновременно</i> . Данная команда блокируется только от запрета управления, выставленного оператором в окне управления.
УП	Вещественный дв. точности	Отображает процент открытия клапана: при <i>УП>Уставка КВО</i> клапан открыт, при <i>УП<Уставка КВЗ</i> клапан закрыт.
Параметр	Вещественный дв. точности	Регулируемый параметр.
Задание	Вещественный дв. точности	Задание, подаваемое на регулятор в режиме <i>Каскад</i> .
Коррекция	Вещественный дв. точности	Значение, используемое для формирования значения рассогласования.
Запрет Больше	Логический	Команда. Если <i>Истина</i> , то блокируются управляющие импульсы в сторону <i>Больше</i> , вырабатываемые в режиме <i>Автомат</i> или <i>Каскад</i> .
Запрет Меньше	Логический	Команда. Если <i>Истина</i> , то блокируются управляющие импульсы в сторону <i>Меньше</i> , вырабатываемые в режиме <i>Автомат</i> или <i>Каскад</i> .
Вкл Дистанция	Логический	Команда на переключение регулятора в режим <i>Дистанция</i> .
Вкл Автомат	Логический	Команда на переключение регулятора в режим <i>Автомат</i> .
Вкл Каскад	Логический	Команда на переключение регулятора в режим <i>Каскад</i> .
Вкл Сквозной	Логический	Команда на переключение регулятора в режим <i>Сквозной</i> .
Насквозь	Вещественный дв. точности	Значение управляющего воздействия от внешнего входа. Используется при переходе регулятора в режим <i>Сквозной</i> .
Авария	Логический	Авария от внешнего входа. Если <i>Истина</i> , на панели состояния диагностики загорается индикатор <i>Авария от внешнего входа</i> .
Местное управление	Логический	Сигнал, блокирующий управление ИМ, и сигнализирующий о переводе его в режим местного управления.
Кр	Вещественный дв. точности	Коэффициент пропорциональной связи.
Td	Вещественный дв. точности	Постоянная времени дифференцирования. Если равна 0, то дифференциальная часть отключается.
Ti	Вещественный дв. точности	Постоянная времени интегрирования. Если равна 0, то интегральная часть отключается.
Квитирование аварии	Логический	Снимает мигание сигнальной области и аварии с фиксацией (например <i>Недостоверность параметра</i>).

Входы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-хбайтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма (представим в десятичном виде): 1 – открыт; 2 – закрыт; 5 – остановлен в промежуточном состоянии. Второй байт представляет состояние режима механизма (представим в двоичном виде): 0x00000001 – дистанция; 0x00000010 – автомат; 0x00000100 – каскад; 0x00001000 – аварийный переход в дистанцию; 0x00010000 – насквозь; Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000010 – авария (от внешнего входа, недостоверность параметра); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте - 262144) авария (второй бит в третьем байте - 131072) при открытом состоянии клапана (состояние 1 в первом байте) и регуляторе в режиме дистанция (первый бит во втором байте - 256). На выходе будет (262144+131072+256+1) 393473.
Выход Авария	Логический	Принимает значение <i>Истина</i> при наличии какой-либо аварии.
Запрет управления	Логический	Принимает значение <i>Истина</i> , если в окне управления нажата кнопка <i>Запретить управление</i> , либо произошла <i>Авария</i> . При этом все команды, подаваемые на входы ФБ, игнорируются.
Управление	Целый	Длительность управляющего сигнала в миллисекундах (используется только в МФК3000).
Каскад	Логический	Принимает значение <i>Истина</i> при работе регулятора в режиме <i>Каскад</i> .
Автомат	Логический	Принимает значение <i>Истина</i> при работе регулятора в режиме <i>Автомат</i> .
Дистанция	Логический	Принимает значение <i>Истина</i> при работе регулятора в режиме <i>Дистанция</i> .
Сквозной	Логический	Принимает значение <i>Истина</i> при работе регулятора в режиме <i>Сквозной</i> .
Период	Беззнаковый целый	Принимает значение, равное периоду опроса собственного цикла объекта, которому принадлежит данный ФБ.
Аварийный переход в дистанцию	Логический	Принимает значение <i>Истина</i> при поступлении команд от технологических защит и возникновении аварии (например, <i>Аппаратной недостоверности</i>).
Error	Вещественный дв. точности	Значение сигнала рассогласования: <i>Задание - Параметр + Коррекция</i> , с учетом выбранного типа зоны нечувствительности.
Task	Вещественный дв. точности	Значение параметра <i>Задание</i> с учетом режима работы регулятора.
p	Вещественный дв. точности	Вспомогательные выходы для определения внутренних составляющих ПДД2-закона.
d	Вещественный дв. точности	
dd	Вещественный дв. точности	
pdd	Вещественный дв. точности	

Символ динамического элемента регулирующего клапана состоит из двух элементов: собственно самого клапана, состоящего из двух треугольников соединенных вершинами, индикатора наличия привода управления от регулятора и поля индикации положения клапана в виде:



Возможные варианты отображения состояния регулирующего клапана и его регулятора:

Состояние	Поведение
РК закрыт, регулятор в <i>Дистанция</i> (На странице свойств <i>Настройки</i> установлено: <i>Использовать Автомат = Истина</i> или <i>Использовать Каскад = Истина</i>).	Ровный белый цвет
РК закрыт, регулятор в <i>Автомат</i> или <i>Каскад</i> (<i>Использовать Автомат = Истина</i> или <i>Использовать Каскад = Истина</i>).	

Состояние

Поведение

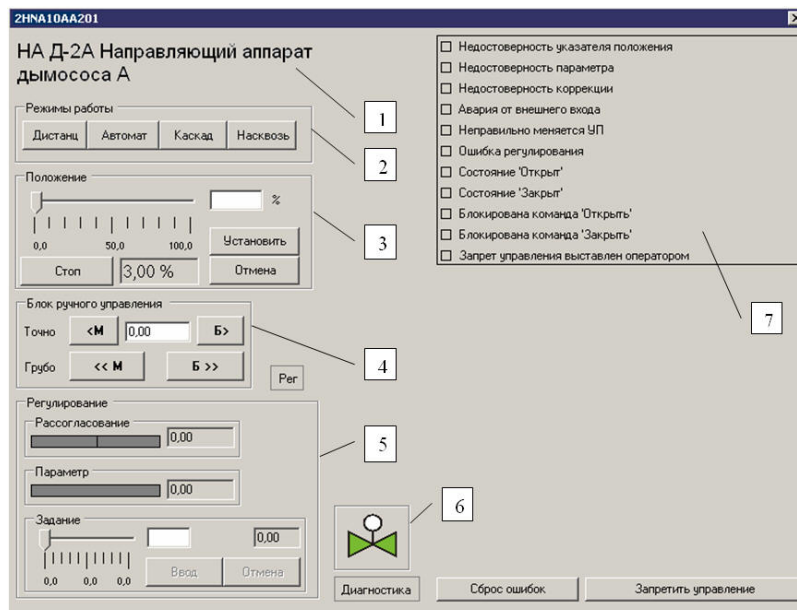
РК закрыт, регулятор в Каскад (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК закрыт, регулятор в Автомат (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК закрыт, регулятор в Дистанция (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК закрыт, регулятор – аварийный переход в Дистанция (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК закрыт, регулятор - аварийный переход в Дистанция (Использовать Автомат = Истина или Использовать Каскад = Истина).
 РК открыт, регулятор в Дистанция (Использовать Автомат = Истина или Использовать Каскад = Истина).
 РК открыт, регулятор в Автомат (Использовать Автомат = Истина или Использовать Каскад = Истина).
 РК открыт, регулятор в Каскад (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК открыт, регулятор в Автомат (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК открыт, регулятор в Дистанция (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК открыт, регулятор – аварийный переход в Дистанция (Использовать Автомат = Истина и Использовать Каскад = Истина).
 РК открыт, регулятор - аварийный переход в Дистанция (Использовать Автомат = Истина или Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор в Дистанция (Использовать Автомат = Истина или Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор в Автомат (Использовать Автомат = Истина или Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор в Каскад (Использовать Автомат = Истина и Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор в Автомат (Использовать Автомат = Истина и Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор в Дистанция (Использовать Автомат = Истина и Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор – аварийный переход в Дистанция (Использовать Автомат = Истина и Использовать Каскад = Истина).
 Недостовверное состояние положения РК, регулятор – аварийный переход в Дистанция (Использовать Автомат = Истина или «Использовать Каскад» = Истина).
 Присутствует сигнал аварии.

Ровный зеленый цвет

Выставлен запрет управления РК оператором.

Мигает красная рамочка.
 Зачеркнут красным крестом.
 Запрещены все операции, включая управление от защит, блокировок, АСР.

При клике мышью на динамическом элементе регулирующего клапана вызывается окно управления в виде:



В поле окна:

- (1) Краткий технологический код и полное наименование механизма;
- Кнопки (2) задания режима работы регулятора;
- Слайдер (3) установки РК в заданное положение для быстрого управления клапаном мышкой и окно ввода значения УП через клавиатуру. При нажатии на кнопку *Установить* производится позиционирование клапана в заданное положение. Причем, при недостоверности сигнала указателя положения слайдер не работает. Кнопка *Стоп* отменяет все управляющие воздействия от оператора (в том числе и от блока ручного управления);
- Блок ручного управления (БРУ) (4) РК для управления перемещением клапана непосредственно оператором. Кнопки группы *Точно* используются для более тонкой настройки. Кнопки группы *Грубо* - для более грубой. Причем, последние рекомендуется использовать для быстрого перемещения ИМ на большие дистанции. Например, из положения *Открыт* в *Закрыт*, когда датчик УП выведен из строя;
- Блок регулирования (5), в котором оператор имеет возможность устанавливать значения задания и отслеживать величину параметра. После установления нового значения задания необходимо нажать кнопку *Ввод*;
- Мнемоническое изображение (6) состояния РК и регулятора;
- Панель диагностики (7) состояния РК. Вызывается при нажатии кнопки *Диагностика*.

Панель диагностики содержит индикаторы:

- *Недостоверность указателя положения* – входной сигнал от указателя положения имеет плохой признак качества;
- *Недостоверность параметра* - сигнал параметра, поступающего на РК, имеет плохой признак качества;
- *Недостоверность коррекции* – входной сигнал коррекции, поступающий на РК, имеет плохой признак качества;
- *Авария от внешнего входа* - вход *Авария* = *Истина*;
- *Неправильно меняется УП*;
- *Ошибка регулирования* – возникает, если значение рассогласования превышает Допустимое отклонение в течение *Контрольного времени регулирования*;
- *Состояние Открыт* - $УП > \text{Уставка КВО}$;
- *Состояние Закрыт* - $УП < \text{Уставка КВЗ}$;
- *Блокирована команда Открыть* - вход *Запрет Открыть* = *Истина*;
- *Блокирована команда Закрыть* - вход *Запрет Закрыть* = *Истина*;
- *Запрет управления выставлен оператором* - в окне управления нажата кнопка *Запретить управление*.

Страница свойств *Настройки ФБ Регулирующий клапан с импульсным управлением* выглядит следующим образом:

Общие Архив Опрос Сообщения Настройки Права доступа	
1 Настройки клапана	
1.1 Время полного хода клапана	100
1.1 Минимальная длительность управляющего импульса (мс)	300
1.2 Уставка КВЗ (% УП)	2
1.3 Уставка КВО (% УП)	98
1.5 Способ формирования выхода	Без строба и без ограничения длительности импульса
2 Дистанционное управление	
2.1 Использовать блокировку закрытия	Истина
2.2 Использовать блокировку открытия	Истина
2.3 Число итераций при установке в положение	3
2.4 Точность установки в положение (%)	2
2.5 Выдержка между итерациями при установке в положение (с)	1
3 Режимы	
3.1 Использовать Автомат	Истина
3.2 Использовать Каскад	Истина
3.3 Использовать Насквозь	Ложь
3.4 Использовать вход 'Местное управление'	Истина
4 Регулирование	
4.1 Кратность пересчета регулятора (ед.)	10
4.10 Максимальная скорость изменения рассогласования (ед.)	0
4.2 Верхний предел регулирования (ед. параметра)	100
4.3 Нижний предел регулирования (ед. параметра)	0
4.4 Максимальное значение рассогласования (%)	100
4.5 Компенсация люфта при закрытии (с)	0
4.6 Компенсация люфта при открытии (с)	0
4.7 Балансировка задания регулятора	Не используется
4.8 Время динамической балансировки (с)	10
4.9 Ограничивать управляющее воздействие при регулировании	Периодом пересчета регулятора
5 Обработка параметра	
5.1 Постоянная времени фильтра параметра (с)	0
5.2 Тип зоны нечувствительности	Тип 2
5.3 Величина полузоны нечувствительности (ед. параметра)	0
6 Диагностика	
6.1 Выдержка при недостоверности параметра (с)	1
6.2 Диагностировать застревание клапана при движении	Ложь
6.3 Контрольное время для диагностики застревания (с)	10
6.4 Минимальное перемещение при диагностике застревания (ед. параметра)	10
6.5 Диагностировать ошибку регулирования	Ложь
6.6 Требуемая точность регулирования (%)	10
6.7 Контрольное время регулирования (с)	100
1.1 Время полного хода клапана Постоянная времени ИМ	
<input type="button" value="Применить"/> <input type="button" value="Отменить"/> <input type="button" value="Справка"/>	

Функции элементов управления группы *Настройка клапана*:

- *Время полного хода клапана* - представляет собой полное время хода исполнительного механизма и используется для расчета управляющих импульсов, как при регулировании, так и при ручном управлении;
- *Минимальная длительность управляющего импульса, (мс)* - минимальная длительность управляющего импульса, которую может почувствовать ИМ;
- *Уставка КВЗ* - процентное значение превышения, которое сигналом УП воспринимается как состояние *Закрыт*;
- *Уставка КВО* - процентное значение превышения, которое сигналом УП воспринимается как состояние *Открыт*;
- *Способ формирования выхода*:
 - Без строба и без ограничения длительности импульса;
 - Без строба с ограничением длительности периодом ФБ;
 - Со стробом - метод выделения некоторого интервала на временной оси, шкале частот и т.п. для увеличения вероятности обнаружения полезных сигналов на фоне помех.

Функции элементов управления группы *Дистанционное управление*:

- *Использовать блокировку закрытия* - использовать/ не использовать вход *Запрет Закрыть*;
- *Использовать блокировку открытия* - использовать/ не использовать вход *Запрет Открыть*;
- Число итераций при установке в положение;
- Точность установки в положение (%);
- Выдержка между итерациями при установке в положение (с).

Функции элементов управления группы *Режимы*:

- *Использовать Автомат* - использовать режим *Автомат*. Отключение режима *Автомат* блокирует соответствующую кнопку в окне управления оператора;

- *Использовать Каскад* - использовать режим *Каскад*. Отключение режима *Каскад* блокирует соответствующую кнопку в окне управления оператора;
- *Использовать Насквозь* - использовать режим *Насквозь*;
- *Использовать вход Местное управление* - определяет использовать/не использовать вход *Местное управление*. При наличии сигнала на данном входе все управление данным ФБ блокируется как от сигнала *Запрет управления*.

Функции элементов управления группы *Регулирование*:

- *Кратность пересчета регулятора* - определяется как произведение периода опроса ФБ на кратность опроса ФБ и на кратность выдачи ШИМ;
- *Максимальная скорость изменения рассогласования*;
- *Верхний предел регулирования* - максимальное значение задатчика;
- *Нижний предел регулирования* - минимальное значение задатчика;
- *Максимальное значение рассогласования* - максимальное по абсолютному значению рассогласование пропускаемое на регулятор. Если рассогласование превышает это значение, рассогласование принимается равным этому значению;
- *Компенсация люфта при закрытии (с)* - длительность компенсационного импульса, выдаваемого при смене движения регулятора на закрытие. Учитывается только в режимах управления *Каскад* или *Автомат*;
- *Компенсация люфта при открытии (с)* - длительность компенсационного импульса, выдаваемого при смене движения регулятора на открытие. Учитывается только в режимах управления *Каскад* или *Автомат*;
- *Балансировка задания регулятора*;
- *Время динамической балансировки*;
- Ограничивать управляющее воздействие при регулировании.

Функции элементов управления группы *Диагностика*:

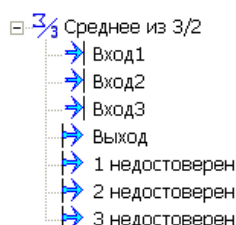
- *Выдержка при недостоверности параметра*;
- *Диагностировать застревание клапана при движении*;
- *Контрольное время для диагностики застревания* - в режиме дистанционного управления во время движения при не изменении УП в течение контрольного времени фиксируется ошибка *неправильно изменяется УП*;
- *Минимальное перемещение при диагностике застревания* - в режиме дистанционного управления во время движения при не изменении УП в течение контрольного времени на заданное приращение фиксируется ошибка *неправильно изменяется УП*;
- *Диагностировать ошибку регулирования* - определяет наличие контроля правильности регулирования;
- *Требуемая точность регулирования (%)* - если за контрольное время регулирования Параметр не попадет в зону задание \pm допустимое отклонение, то выдается сигнал *Авария*;
- *Контрольное время регулирования (с)* - время, через которое ФБ выдаст сигнал *Авария*, если сигнал *Параметр* не войдет в зону допустимого отклонения от задания.

Функции элементов управления группы *Обработка параметра*:

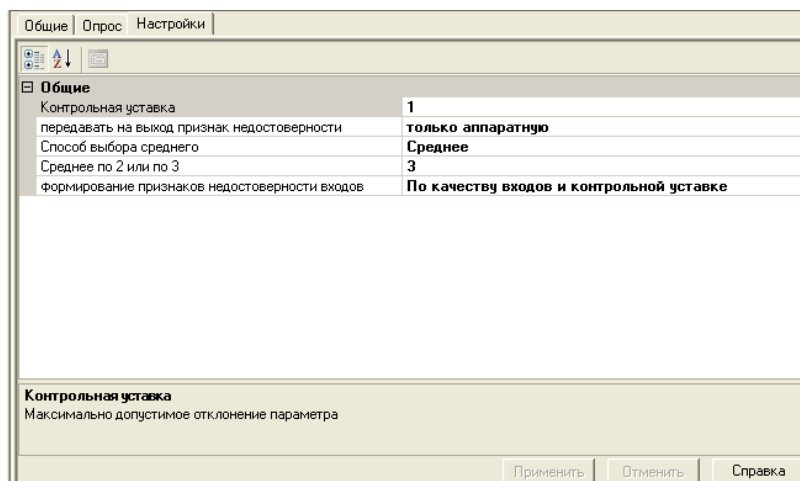
- *Постоянная времени фильтра параметра (с)* - настроечный параметр фильтра в секундах;
- *Тип зоны нечувствительности* - используется при пересчете сигнала рассогласования. Включает 2 типа функций, вычисляемых по формулам описанным ранее;
- Величина полузоны нечувствительности.

Средние из 3/2

Вид ФБ в дереве Объекта:



Страница свойств *Настройки* ФБ выглядит следующим образом:

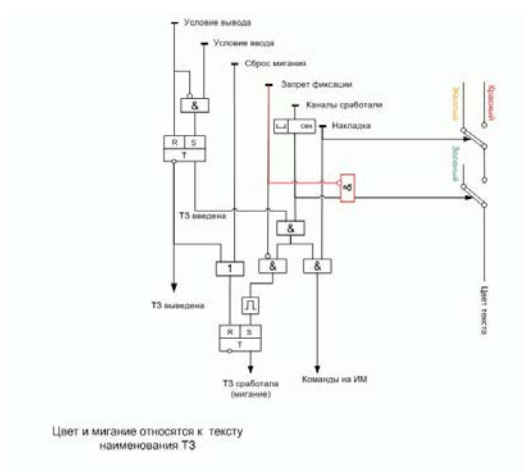


Технологическая защита

Вид функционального блока в дереве *Объекта*:

- [-] T3 Технологическая защита
 - Ввести защиту
 - Вывести защиту
 - Запрет фиксации
 - Сброс
 - Каналы Т3 сработали
 - Т3 выведена
 - Т3 сработывала (мигание)
 - Т3 введена
 - Каналы Т3 сработали (задержанный для сигнализации)
 - [-] Накладки
 - накладка1
 - накладка2
 - [-] Команды
 - команда1
 - команда2
 - [-] Индикация срабатывания Т3 (цвет текста) (0=нет,1=предупр,2=авар)
 - статус1
 - статус2

Алгоритм работы:



Входы ФБ:

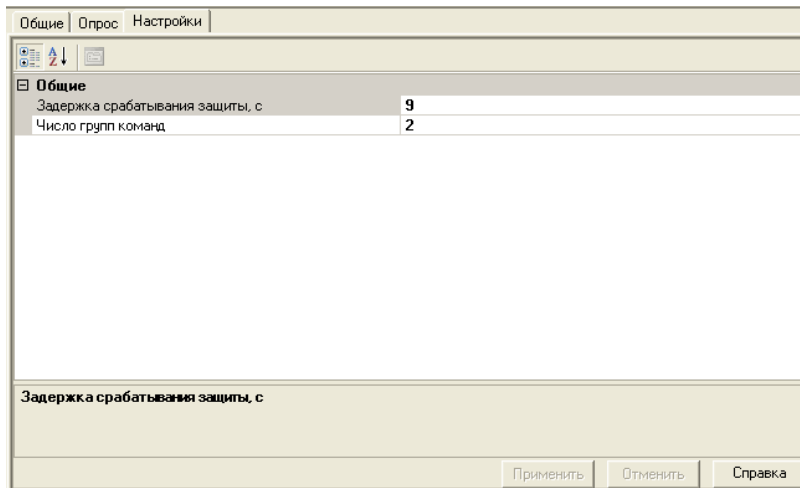
Обозначение	Тип	Описание
Ввести защиту	Логический	Подается выход заданной логической сборки условий ввода Т3.
Вывести защиту	Логический	Подается выход заданной логической сборки условий вывода Т3.
Запрет фиксации	Логический	Служит для запрета сигнализации после срабатывания первой защиты. То есть если защита сработала, следующие за ней защиты не будут индицировать факт срабатывания.
Сброс	Логический	Для сброса сигнализации на выходе Т3 сработывала (сброс мигания)

Каналы ТЗ сработали	Логический	сигнализации на мнемосхеме). Подается выход заданной логической сборки условий срабатывания каналов ТЗ.
Накладки	Логический	Наличие накладок для прохождения команд на соответствующие группы исполнительных механизмов.

Выходы ФБ:

Обозначение	Тип	Описание
ТЗ выведена	Логический	Сигнализация о выведенном состоянии ТЗ.
ТЗ сработывала (мигание)	Логический	Сигнализация о срабатывании ТЗ (мигание цвета текста наименования ТЗ на мнемосхеме).
ТЗ введена	Логический	Защита введена (условие ввода выполнено, а условие вывода нет).
Каналы ТЗ сработали	Логический	Сработали каналы защиты и выдержано время задержки срабатывания защиты.
Команды	Логический	Команды ТЗ, подаваемые на входы исполнительных механизмов. Позволяет изменять цвет текста наименования ТЗ на мнемосхеме и возвращает значение статуса:
Индикация срабатывания ТЗ	Целый	зеленый (статус 0) – защита не сработала; желтый (статус 1) – предупреждение (сработала защита без накладки); красный (статус 2) – авария (сработала защита и введена накладка, на исполнительный механизм подана команда).

Страница свойств *Настройки* ФБ *Технологическая защита* выглядит следующим образом:

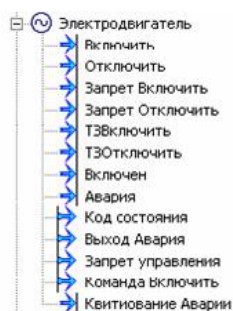


Элементы управления:

- *Задержка срабатывания защиты (с)* - Промежуток времени, который необходимо выдерживать после поступления переднего фронта на вход *Каналы ТЗ сработали*, чтобы избежать ложного срабатывания защит. Если сигнал *Вкл.* в течение этого промежутка времени продолжает присутствовать на входе, то на *выход Каналы ТЗ сработали* тоже подается сигнал *Вкл.*
- *Число групп команд* - пределяет количество входов/выходов в группах переменных *Накладки*, *Команды*, *Индикация срабатывания ТЗ* в дереве объектов.

Электродвигатель

Вид функционального блока в дереве объекта:



Визуальный функциональный блок *Электродвигатель* осуществляет функцию управления электродвигателем механизма собственных нужд. ФБ также предоставляет возможность

отображения на мнемосхеме состояния исполнительного механизма, привязанного к этому ФБ.

При поступлении команды на вход *Включить* на выходе ФБ устанавливается положительный потенциальный сигнал, который переводит клапан во включенное состояние. При поступлении команды на вход *Отключить* снимает выходной потенциальный сигнал, удерживающий клапан во включенном положении. При поступлении команды на вход *Включить*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Отключить* и *Запрет*, выдается команда на открытие ИМ (у мнемонического символа мигает белое поле). По истечении контрольного времени включения/отключения, проверяется исполнение команды. Если команда выполнена, мнемонический символ становится зеленым. При обнаружении неисполнения команды выставляется признак *Не включился за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета.

При поступлении команды на вход *Отключить*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Включить* и *Запрет*, выдается команда на закрытие ИМ (у мнемонического символа мигает зеленое поле). По истечении контрольного времени включения/отключения, проверяется исполнение команды. Если команда выполнена, мнемонический символ становится белым. При обнаружении неисполнения команды выставляется признак *Не отключился за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета.

Возможности управления:

- Автоматическое, через управляющие входы;
- Ручное, через окно операторского управления.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

Мнемоническое отображение функционального блока на мнемосхеме:

Состояние	Поведение
Электродвигатель <i>Включен</i>	Ровный зеленый цвет.
Электродвигатель <i>Отключен</i>	Ровный белый цвет.
Недостовверное состояние	Ровный серый цвет.
Состояние включения	При переключении из состояния <i>Отключен</i> в <i>Включен</i> мигает белое на сером поле.
Состояние отключения	При переключении из состояния <i>Включен</i> в <i>Отключен</i> , мигает зеленое на сером поле.
Присутствует сигнал аварии	Мигает красная рамочка.
<i>Включен</i> и запрет включения	Запрещены все операции, включая управление от защит и блокировок.
<i>Отключен</i> и запрет включения	Запрещены все операции, включая управление от защит и блокировок.
Недостовверное состояние и запрет включения	Запрещены все операции, включая управление от защит и блокировок.

Элемент динамический, при клике вызывается окно управления:

По кнопке *Диагностика* вызывается панель состояния диагностики электродвигателя. Панель диагностики содержит индикаторы:

- *Аппаратная недостоверность* - входные сигналы, отображающие состояние электродвигателя, имеют плохой признак качества;
- *Не включился за контрольное время* - в случае отсутствия сигнала *Истина* на входе «Включен» по истечении контрольного времени;
- *Не отключился за контрольное время* - в случае отсутствия сигнала *Ложь* на входе *Включен* по истечении контрольного времени;
- *Авария от внешнего входа* – вход *Авария* = *Истина*;
- *ТЗ Включить* и *ТЗ Отключить одновременно* - в случае одновременного поступления сигнала *Истина* на входы *ТЗОтключить* и *ТЗВключить*, имеющих одинаковый приоритет;
- *Несанкционированное отключение двигателя* – появление сигнала *Ложь* на входе *Включен*, при отсутствии команд на отключение от оператора, ФГУ или технологических защит;
- *Несанкционированное включение двигателя* – появление сигнала *Истина* на входе *Включен*, при отсутствии команд на включение от оператора, ФГУ или технологических защит;

- *Сработало реле Включен - вход Включен» = Истина;*
- *Блокирована команда Включить - вход Запрет Включить = Истина;*
- *Блокирована команда Отключить - вход Запрет Отключить = Истина;*
- *Запрет управления выставлен оператором - в окне управления нажата кнопка Запретить управление.*

Входы ФБ:

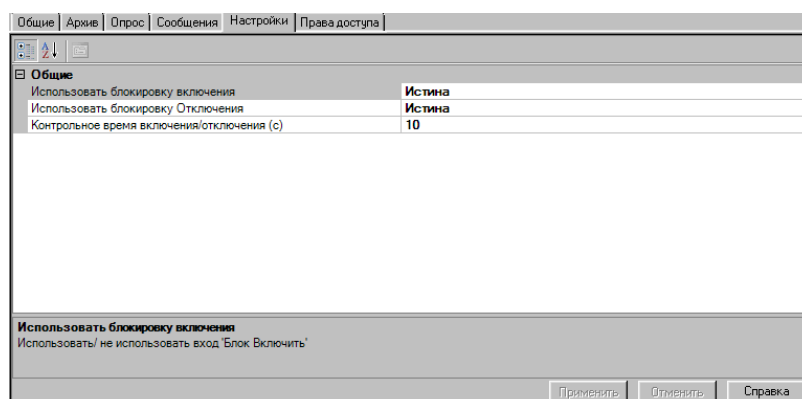
Имя	Тип	Примечание
Включить	Логический	Команда. Переводит выход <i>Команда Включить</i> в состояние <i>Истина</i> при отсутствии сигнала <i>Запрет Включить</i> и команд от технологических защит. В случае одновременного поступления с командой <i>Отключить</i> , имеющей тот же приоритет, обе команды игнорируются.
Отключить	Логический	Команда. Переводит выход <i>Команда Включить</i> в состояние <i>Ложь</i> при отсутствии сигнала <i>Запрет Отключить</i> и команд от технологических защит. В случае одновременного поступления с командой <i>Включить</i> , имеющей тот же приоритет, обе команды игнорируются.
Запрет Включить	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Включить</i> от входа и от оператора и на панели состояния диагностики загорается индикатор <i>Блокирована команда Включить</i> . Данная блокировка не оказывает влияние на команду от ТЗ. Вход появляется, если на странице свойств <i>Настройки</i> установлена <i>Истина</i> в графе <i>Использовать блокировку включения</i> .
Запрет Отключить	Логический	Команда. Если <i>Истина</i> , то блокируются команды <i>Отключить</i> от входа и от оператора и на панели состояния диагностики загорается индикатор <i>Блокирована команда Отключить</i> . Данная блокировка не оказывает влияние на команду от ТЗ. Вход появляется, если на странице свойств <i>Настройки</i> установлена <i>Истина</i> в графе <i>Использовать блокировку отключения</i> .
ТЗВключить	Логический	Команда <i>Включить</i> от технологических защит. Имеет повышенный приоритет. В случае одновременного поступления с командой <i>ТЗОтключить</i> , имеющей тот же приоритет, обе команды игнорируются, а на панели состояния диагностики загорается индикатор <i>ТЗ Включить и ТЗ Отключить одновременно</i> . Данная команда блокируется только от запрета управления, выставленного оператором в окне управления.
ТЗОтключить	Логический	Команда <i>Отключить</i> от технологических защит. Имеет повышенный приоритет. В случае одновременного поступления с командой <i>ТЗВключить</i> , имеющей тот же приоритет, обе команды игнорируются, а на панели состояния диагностики загорается индикатор <i>ТЗ Включить и ТЗ Отключить одновременно</i> . Данная команда блокируется только от запрета управления, выставленного оператором в окне управления.
Включен	Логический	Определяет текущее состояние электродвигателя.
Авария	Логический	Авария от внешнего входа. Если <i>Истина</i> , на панели состояния диагностики загорается индикатор <i>Авария от внешнего входа</i> .
Квитирование Аварии	Логический	Снимает мигание сигнальной области и аварии с фиксацией (например <i>Не включился за контрольное время</i>).

Входы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-хбайтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма: 1 – включен; 2 – отключен; 3 – включается; 4 – отключается; 8 – недостоверное состояние. Второй байт представляет состояние режима механизма и для данного ФБ не используется. Третий байт – флаги (представим в двоичном виде): 0x00000001 – запрет управления; 0x00000010 – авария (от внешнего входа, не включился за контрольное время); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте - 262144) авария (второй бит в третьем байте - 131072) при включенном состоянии электродвигателя (262144+131072+1). На выходе будет 393217.
Выход Авария	Логический	Принимает значение <i>Истина</i> при наличии какой-либо аварии.
Запрет управления	Логический	Принимает значение <i>Истина</i> , если в окне управления нажата кнопка <i>Запретить управление</i> , либо произошла <i>Авария</i> . При этом все команды, подаваемые на входы ФБ, игнорируются.

Имя	Тип	Примечание
Команда Включить	Логический	Потенциальная команда, выдаваемая на исполнительные цепи.

Страница свойств *Настройки* ФБ *Электродвигатель* выглядит следующим образом:

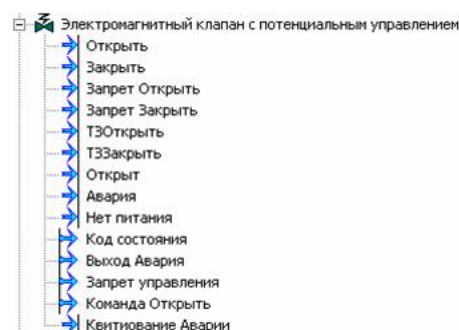


Функции элементов управления группы *Общие*:

- *Использовать блокировку включения* – использовать/ не использовать вход *Запрет Включить*;
- *Использовать блокировку отключения* – использовать/ не использовать вход *Запрет Отключить*;
- *Контрольное время включения/ отключения (с)* – при не переключении ИМ по истечению данного времени формируется сигнал *Авария*.

Электромагнитный клапан с потенциальным управлением

Вид функционального блока в дереве *Объекта*:



Функциональный блок *Электромагнитный клапан с потенциальным управлением*, загружаемый в контроллер или исполняемый в среде MasterSCADA, осуществляет функцию управления двухпозиционным исполнительным механизмом. ФБ также предоставляет возможность отображения на видеокадре мнемонического символа состояния исполнительного механизма, привязанного к этому ФБ.

При поступлении команды на вход *Открыть* на выходе ФБ устанавливается положительный потенциальный сигнал, который переводит клапан в открытое состояние. При поступлении команды на вход *Закреть* снимает выходной потенциальный сигнал, удерживающий клапан в открытом положении. При поступлении команды на вход *Открыть*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Закреть* и *Запрет*, выдаётся команда на открытие ИМ (выход *Открыть* устанавливается в лог. 1, мнемонический символ окрашивается в зеленый цвет). По истечении контрольного времени включения/отключения, проверяется исполнение команды. При обнаружении неисполнения команды выставляется признак *Не открылся за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета. При поступлении команды на вход *Закреть*, отсутствия запрета управления, выставленного оператором, отсутствия команды *Открыть* и *Запрет*, выдаётся команда на открытие ИМ (выход *Открыть* устанавливается в лог. 0, мнемонический символ окрашивается в белый цвет). По истечении контрольного времени включения/отключения, проверяется исполнение команды. При обнаружении неисполнения коман-

ды выставляется признак *Не закрылся за контрольное время*. Мнемонический символ обрамляется рамкой красного цвета.

Возможности управления:

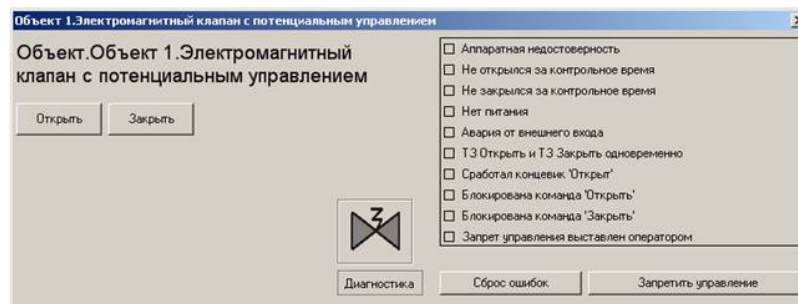
- Автоматическое, согласно логике алгоритма;
- Дистанционное, которое позволяет оператору непосредственно задавать управляющее воздействие.

Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

Символ динамического элемента электромагнитного клапана состоит из двух независимых динамических элементов в виде треугольников с соединенными вершинами и изображением в рассечке пружины. При горизонтальном расположении на трубопроводе левый сегмент символизирует закрытое состояние, а правый – открытое. При вертикальном расположении, нижний сегмент – закрытое, верхний – открытое.

Состояние	Поведение
Закрыто	Ровный белый цвет.
Открыто	Ровный зеленый цвет.
Промежуточное положение	При движении из положения <i>Открыт = Ложь</i> в <i>Открыт = Истина</i> мигает зеленое поле, из <i>Открыт = Истина</i> в <i>Открыт = Ложь</i> мигает белое поле.
Присутствует сигнал аварии	Мигает красная рамочка.
Недостовверное состояние	Ровный серый цвет.
Закрыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Открыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Недостовверное состояние и запрет управления	Запрещены все операции, включая управление от защит и блокировок.

Элемент динамический. При клике вызывается окно управления:



По кнопке *Диагностика* вызывается панель состояния диагностики выбранного механизма. Панель диагностики содержит индикаторы:

- *Аппаратная недостоверность* - входные сигналы, отображающие состояние электродвигателя, имеют плохой признак качества;
- *Не открылся за контрольное время* - в случае отсутствия сигнала *Истина* на входе *Открыт* по истечении контрольного времени;
- *Не закрылся за контрольное время* - в случае отсутствия сигнала *Ложь* на входе *Закрыт* по истечении контрольного времени;
- *Нет питания* – на вход ФБ одновременно пришли два взаимоисключающих сигнала *Открыт = Истина* и *Закрыт = Истина*, либо вход *Нет питания = Истина* в течение контрольного времени открытия/закрытия;
- *Авария от внешнего входа* - вход *Авария = Истина*;
- *ТЗ Открыть и ТЗ Закрыть одновременно* - в случае одновременного поступления сигнала *Истина* на входы *ТЗОткрыть* и *ТЗЗакрыть*, имеющих одинаковый приоритет;
- *Сработал концевик Открыт* - вход *Открыт = Истина*;
- *Блокирована команда Открыть* - вход *Запрет Открыть = Истина*;
- *Блокирована команда Закрыть* - вход *Запрет Закрыть = Истина*;
- *Запрет управления выставлен оператором* - в окне управления нажата кнопка *Запретить управление*.

Входы ФБ:

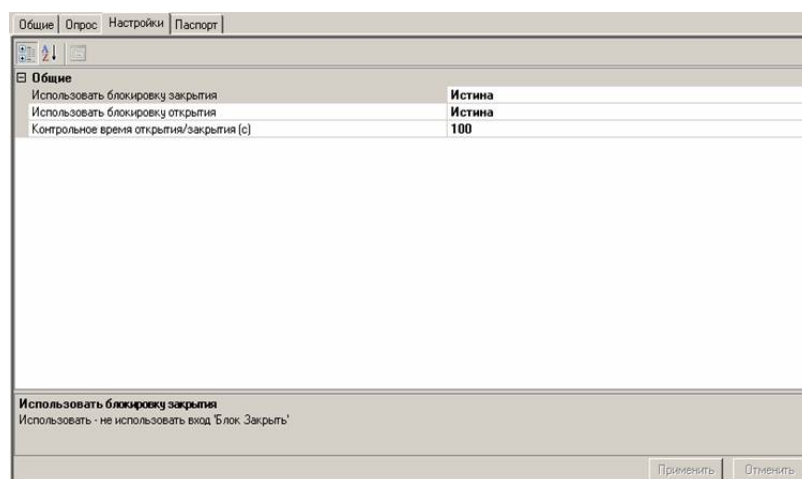
Имя	Тип	Примечание
-----	-----	------------

Имя	Тип	Примечание
Открыть	Логический	Происходит открытие клапана, если он был закрыт.
Закрыть	Логический	Происходит закрытие клапана, если он был открыт.
Запрет Открыть	Логический	Блокировка открытия клапана.
Запрет Закрыть	Логический	Блокировка закрытия клапана.
T3Открыть	Логический	Команда от ТЗ, управляющая открытием клапана. Данная команда имеет повышенный приоритет. Работа этой команды не зависит от запретов по входам управления. Блокируется только оператором через окно управления.
T3Закрыть	Логический	Команда от ТЗ, управляющая закрытием клапана. Данная команда имеет повышенный приоритет. Работа этой команды не зависит от запретов по входам управления. Блокируется только оператором через окно управления.
Открыт	Логический	Флаг открытого состояния клапана.
Авария	Логический	Внешний аварийный сигнал (резервный), используемый разработчиком.
Нет питания	Логический	На вход ФБ одновременно пришли два взаимоисключающих сигнала <i>Открыт = Истина</i> и <i>Закрыт = Истина</i> , либо вход <i>Нет питания = Истина</i> в течение контрольного времени открытия/закрытия.
Квитирование Аварии	Логический	Внешняя команда квитирования ошибок.

Выходы ФБ:

Имя	Тип	Примечание
Код состояния	Беззнаковый целый	Обобщенный сигнал состояния ИМ.
Выход Авария	Логический	Обобщенный сигнал <i>Авария</i> о возникших ошибках в работе ИМ.
Запрет управления	Логический	Блокирует выдачу каких-либо команд из функционального блока. Эта команда обладает наивысшим приоритетом.
Команда Открыть	Логический	Команда, выдаваемая на исполнение во внешние исполнительные цепи.

Страница свойств *Настройки* ФБ *Электромагнитный клапан с потенциальным управлением* выглядит следующим образом:



Функции элементов управления группы *Общие*:

- *Использовать блокировку закрытия* – использовать/ не использовать вход *Блок Закрыть*;
- *Использовать блокировку открытия* – использовать/ не использовать вход *Блок Открыть*;
- *Контрольное время открытия/ закрытия (с)* – при не срабатывании ИМ по истечению данного времени формируется сигнал *Авария*.

Электромагнитный клапан без управления

Вид функционального блока в дереве объекта:

Функциональный блок *Электромагнитный клапан без управления*, загружаемый в контроллер, или исполняемый в среде MasterSCADA, осуществляет функцию контроля состояния двухпозиционного исполнительного механизма. ФБ также предоставляет возможность отображения на видеокадре мнемоническим символом состояния ИМ, привязанного к этому ФБ.

При поступлении логического сигнала с концевика открытого положения ИМ на выходе ФБ устанавливается флаг, сигнализирующий о положении ИМ *Открыт*. При поступлении логического сигнала с концевика закрытого положения ИМ на выходе ФБ устанавливается флаг, сигнализирующий о положении ИМ *Закрыт*. При поступлении логического сигнала с концевика *Открыт*, отсутствия сигнала на входе *Закрыт*, на выходе ФБ устанавливается флаг *Открыт* (мнемонический символ окрашивается в зеленый цвет). По истечении контрольного времени включения/отключения и наличии признака равнозначности (пришли одинаковые сигналы с концевиков), на выходе ФБ устанавливается сигнал Недостоверность. Мнемонический символ окрашивается в серый цвет. При поступлении логического сигнала с концевика *Закрыт*, отсутствия сигнала на входе *Открыт*, на выходе ФБ устанавливается флаг *Закрыт* (мнемонический символ окрашивается в белый цвет). По истечении контрольного времени включения/отключения и наличии признака равнозначности (пришли одинаковые сигналы с концевиков), на выходе ФБ устанавливается сигнал *Недостоверность*. Мнемонический символ окрашивается в серый цвет. Возможности управления – отсутствуют. Поддерживаемый тип опроса - любой. Восстановление при рестарте - поддерживается.

Входы и выходы ФБ:

	Имя	Тип	Примечание
Входы	Открыт	Логический	Флаг открытого состояния клапана.
	Закрыт	Логический	Флаг закрытого состояния клапана.
	Нет питания	Логический	На вход ФБ одновременно пришли два взаимоисключающих сигнала <i>Открыт = Истина</i> и <i>Закрыт = Истина</i> , либо вход <i>Нет питания = Истина</i> в течение контрольного времени открытия/закрытия.
	Авария	Логический	Внешний аварийный сигнал (резервный), используемый разработчиком.
Выходы	Квитирование Аварии	Логический	Внешняя команда квитирования ошибок.
	Код состояния	Беззнаковый целый	Информирует о текущем состоянии объекта. Представляет собой 4-байтный код (для простоты записывается в десятичном виде). Первый байт представляет состояние работы механизма: 1 – открыт; 2 – закрыт; 8 – недостоверное состояние. Второй байт представляет состояние режима механизма и для данного ФБ не используется. Третий байт – флаги (представим в двоичном виде): 0x00000010 – авария (от внешнего входа, нет питания); 0x00000100 – квитирование аварии. Отображается в виде числа, равного сумме состояний. Например, квитирована (третий бит в третьем байте - 262144) авария (второй бит в третьем байте - 131072) при открытом состоянии клапана (262144+131072+1). На выходе будет 393217.
	Выход Авария	Логический	Обобщенный сигнал <i>Авария</i> о возникших ошибках в работе ИМ
	Запрет управления	Логический	Блокирует выдачу каких либо команд из функционального блока. Эта команда обладает наивысшим приоритетом.

Символ динамического элемента электромагнитного клапана состоит из двух независимых динамических элементов в виде треугольников с соединенными вершинами и изображением в рассечке пружины. При горизонтальном расположении на трубопроводе левый сегмент символизирует закрытое состояние, а правый – открытое. При вертикальном расположении, нижний сегмент – закрытое, верхний – открытое.

Состояние	Поведение
Закрыто	Ровный белый цвет.
Открыто	Ровный зеленый цвет.
Промежуточное положение	При движении из положения <i>Открыт = Ложь</i> в <i>Открыт = Истина</i> мигает зеленое поле, из <i>Открыт = Истина</i> в <i>Открыт = Ложь</i> мигает белое поле.
Присутствует сигнал аварии	Мигает красная рамочка.

Состояние	Поведение
Недостовверное состояние	Ровный серый цвет.
Закрыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Открыто и запрет управления	Запрещены все операции, включая управление от защит и блокировок.
Недостовверное состояние и запрет управления	Запрещены все операции, включая управление от защит и блокировок.

Элемент динамический. При клике вызывается окно управления:

Панель диагностики содержит индикаторы:

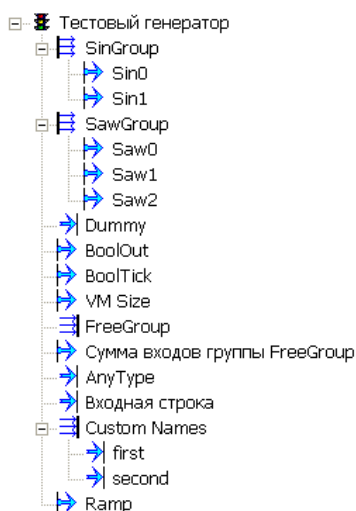
- *Аппаратная недостаточность* - входные сигналы, отображающие состояние электродвигателя, имеют плохой признак качества;
- *Нет питания* – на вход ФБ одновременно пришли два взаимоисключающих сигнала *Открыт = Истина* и *Закрыт = Истина*, либо вход *Нет питания = Истина* в течение контрольного времени открытия/закрытия;
- *Авария от внешнего входа* - вход *Авария = Истина*;
- *Сработал концевик Открыт* - вход *Открыт = Истина*;
- *Сработал концевик Закрыт* - вход *Закрыт = Истина*.

Страница свойств *Настройки* ФБ *Электромагнитный клапан без управления* выглядит следующим образом:

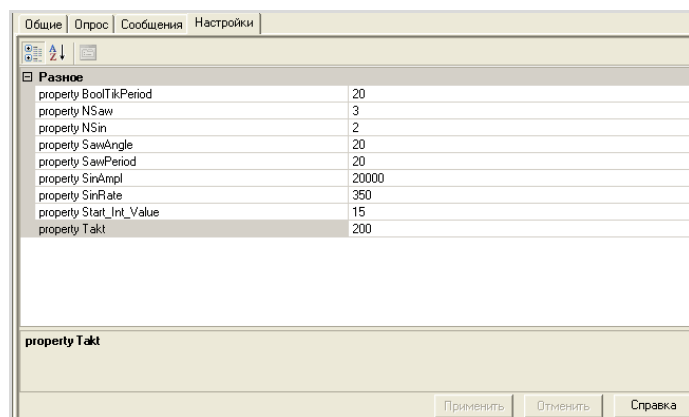
Контрольное время открытия/ закрытия – при несрабатывании ИМ по истечению данного времени формируется сигнал *Авария*.

Тестовый генератор

Вид ФБ в дереве Объекта:



Страница свойств *Настройки* ФБ *Тестовый генератор*:

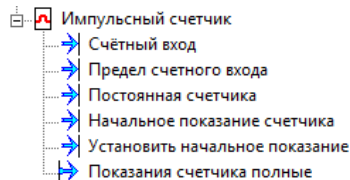


Отраслевая библиотека Электроэнергетика

Импульсный счетчик

Данный ФБ предназначен для вычисления показаний импульсного счетчика (не обязательно электрического). Алгоритм основан на вычислении разности количества импульсов при каждом новом опросе счетного входа и прибавлении этой разности к сохраняемому значению.

Вид функционального блока в дереве объекта:

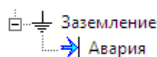


Входы и выходы ФБ:

	Имя	Тип	Примечание
Входы	Счетный вход	Беззнаковый целый	Количество импульсов полученных от электросчётчика счётным входом дискретного модуля. Минимальное значение равно 0, максимальное значение определяется разрядностью счётного входа дискретного модуля. Допустим, в модуле МВ110-16Д, счётный вход 16 битный, то есть его значение лежит в пределах от 0 до 65535 включительно. При достижении максимального значения счётный вход обнуляется и счёт продолжается далее.
	Предел счетного входа	Вещественный дв.точности	Максимальное значение количества импульсов счётного входа дискретного модуля (и переменной <i>Счётный вход</i>), после которого этот счётный вход обнуляется. Для модуля МВ110-16Д это 65535, в других модулях дискретного ввода могут стоять более ёмкие счётные входы, например I-7080, в котором они 32 разрядные.
	Постоянная счётчика	Вещественный дв.точности	Количество импульсов, которое выдаёт электросчётчик, при прохождении через него 1 кВт*ч электроэнергии. Обычно <i>Постоянная счётчика</i> обозначается на корпусе счётчика и в его паспорте в виде <i>1000 имп/кВт*ч</i> (в примере со счётчиком Меркурий 230).
	Начальное показание счетчика	Вещественный дв.точности	Количество электроэнергии в кВт*ч указанное на счётчике при его установке. Задаётся при установке нового счётчика. Необходимо для полного соответствия показаний электросчётчика с вычисленным из количества импульсов значением.
	Установить начальное показание счетчика	Логический	Дискретная значение установки начального показания счётчика после его установки. Передаёт значение переменной <i>Начальное показание счётчика</i> в расчёты ФБ и обнуляет значение количества посчитанной электроэнергии прошлым счётчиком.
Выходы	Показания счетчика полные	Вещественный дв.точности	Количество электроэнергии в кВт*ч пройденное через счётчик, вычисленное по количеству переданных счётчиком импульсов. Состоит из начального показания счётчика и значением посчитанной этим счётчиком электроэнергии после его установки

Заземление

Данный элемент не является визуальным функциональным блоком и отображается только на мнемосхеме. Он имеет ряд специальных свойств которые могут быть настроены проектировщиком, а также задинамизированы при помощи редактора мнемосхем. Данный элемент не имеет каких-либо окон оператора, и не обладает логикой для приема команд от оператора. Он служит лишь для отображения состояния. Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается. Вид функционального блока в дереве объекта:



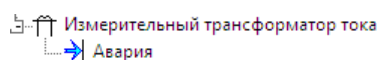
Мнемоническое отображение функционального блока на мнемосхеме:

Изображение	Описание
	Возможность динамизировать цвет линий элемента.
	Возможность подстраивать толщину линий элемента.

Измерительный трансформатор тока

Данный элемент не является визуальным функциональным блоком и отображается только на мнемосхеме. Он имеет ряд специальных свойств которые могут быть настроены проектировщиком, а также задинамизированы при помощи редактора мнемосхем. Данный элемент не имеет каких-либо окон оператора, и не обладает логикой для приема команд от оператора. Он служит лишь для отображения состояния.

Вид функционального блока в дереве объекта:



Мнемоническое отображение функционального блока на мнемосхеме:

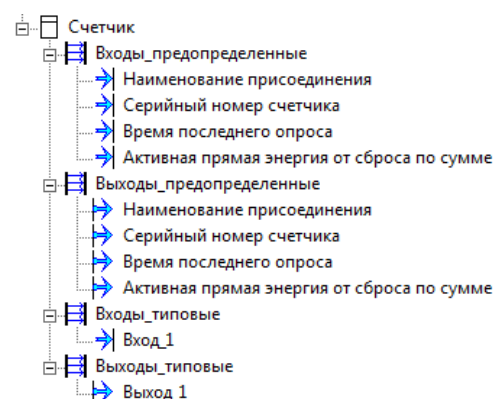
Изображение	Описание
	Возможность динамизировать цвет линий элемента.
	Динамизируемые свойства: <i>Показать рамку, Мигать рамкой, Цвет рамки.</i>
	Возможность подстраивать толщину линий элемента и рамки.

Поддерживаемый тип опроса - любой. Восстановление при рестарте – поддерживается.

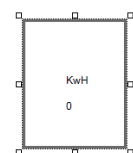
Счетчик

ВФБ *Счётчик* предназначен для отображения на мнемосхеме введенных в него значений в виде отдельной таблицы с возможностью выбрать, какие именно значения отображать. Так же он работает как повторитель, т.е. значения поданный на вход, будут без изменений переданы на выход.

Вид функционального блока в дереве объекта:



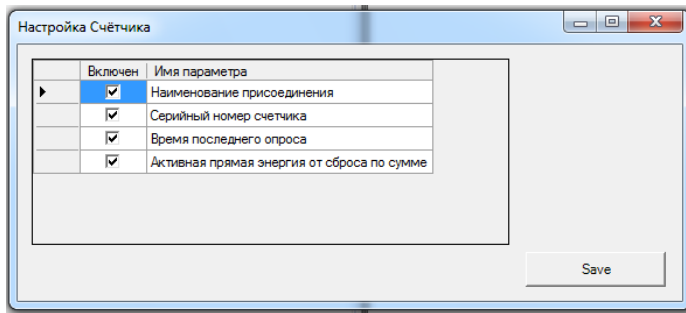
Мнемоническое отображение функционального блока на мнемосхеме *Счетчик*:



Свойства в палитре свойств:

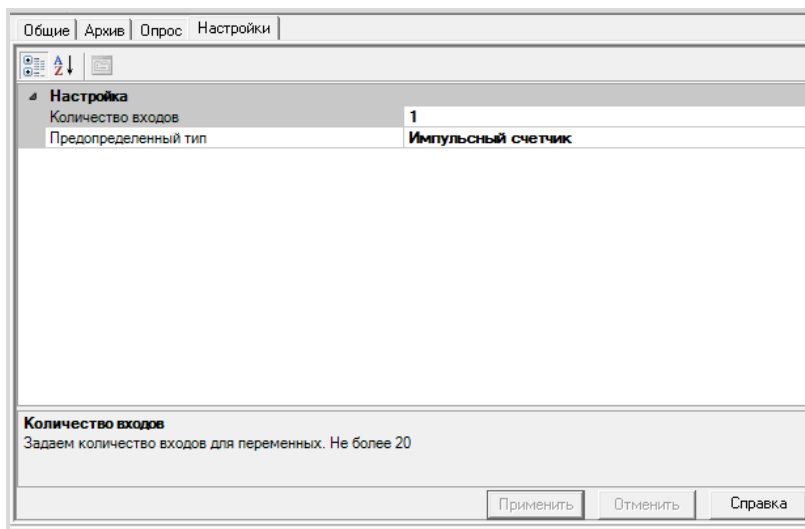
Свойства	Описание
Метка	Текст отображаемый на мнемосхеме, по умолчанию стоит <i>KwH</i> .
Параметр на изображении	Номер входа начиная с нуля, данные которого будут выводиться на изображение ВФБ.
Толщина линии	Обозначает толщину линий ВФБ.
Цвет линии	Цвет отображаемых линий.
Цвет фона	Цвет фона данного ВФБ.

Через контекстное меню, *Свойства* вызывается меню настройки *Счетчика*:



В режиме разработки есть возможность выбрать какие именно переменные будут видны в процессе разработки (снять галочку для того, чтобы не было видно). Так же есть возможность переименовать выбранные переменные. После всех манипуляций, для сохранения изменений нужно нажать *Сохранить*, так же нужно будет согласиться сохранить все изменения на мнемосхеме.

Страница свойств *Настройки* ФБ *Счетчик* выглядит следующим образом:



- ❖ **Важно!** количество входов должно быть не более 20. Настраивает количество типовых входов выходы. Созданные входы выходы будут иметь тип вещественный дв. точности.

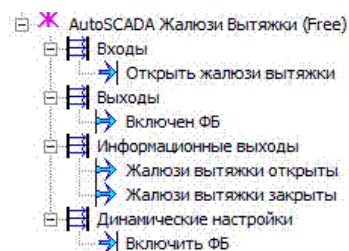
Предопределенный тип - создает именованные входы выходы, с именами и типами переменных, которые наиболее часто используются в соответствующих счетчиках. На текущий момент есть 2 типа:

- Импульсный счетчик;
- Интерфейсный счетчик.

Отраслевая библиотека Вентиляция

Жалюзи вытяжки

Вид визуального функционального блока (ФБ) в дереве объекта:



Визуальный функциональный блок *Жалюзи вытяжки* предназначен для определения состояния заслонок данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ

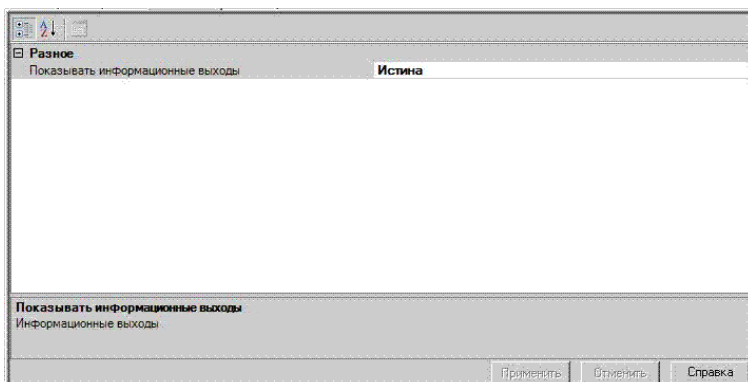
предназначен для визуализации технологических параметров работы жалюзи и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон).

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы жалюзи;
- Отдельное pop-up окно с изображением жалюзи вытяжки, доступное только в режиме RT по нажатию на первое изображение. В данном ФБ отображает только анимацию работы жалюзи.

Логика работы функционального блока следующая - зависимости от комбинации входных сигналов определяется и отображается состояние жалюзи в каждый момент времени.

Страница свойств *Настройки* ФБ жалюзи вытяжки выглядит следующим образом:






Установка флага *Показывать информационные выходы* определяет наличие выходов в дереве, относящихся к отображения состояний жалюзи.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Открыть жалюзи вытяжки	Логический	от OPC	Сигнал на открытие жалюзи. Информирует о текущем состоянии объекта. Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке.
Динамические настройки ФБ	Включить ФБ	Логический	от OPC	Состояние 0 (выкл) данного входа скрывает (делает невидимым) всякое представление жалюзи. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно.

Выходы ФБ:

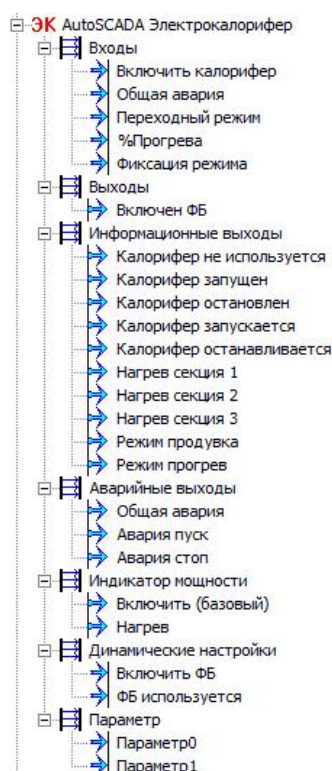
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входов <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Жалюзи вытяжки открыты	Логический	Принимает значение 1 при значении входов <i>Включить ФБ</i> = 1 и <i>Открыть жалюзи вытяжки</i> = 1.	 

Имя	Тип	Зависимости (Формулы, связи)	Примечание
Жалюзи вытяжки закрыты	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1</i> , <i>Открыть жалюзи вытяжки = 0</i>	 <p>Выходной клапан</p> <p>Состояние: Закрыт</p>

- ❖ **Важно!** Наличие группы информационных выходов настраивается на странице свойств ФБ *Настройки*.

Электрокалорифер

Вид функционального блока в дереве Объекта:



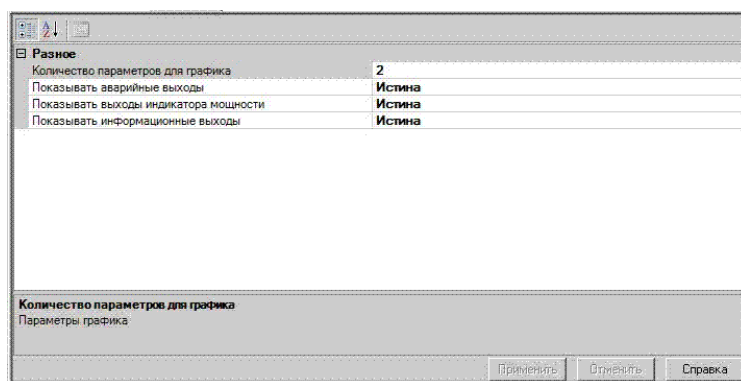
Визуальный функциональный блок *Электрокалорифер* предназначен для определения состояния калорифера данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы калорифера и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Первый – изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы калорифера, аварий и параметра мощности в виде прогресс бара;
- Отдельное рор-уп окно с изображением электрокалорифера, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы калорифера, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы ФБ следующая - В зависимости от комбинации входных сигналов определяется и отображается состояние калорифера в каждый момент времени.

Страница свойств *Настройки* ФБ электрокалорифер выглядит следующим образом:



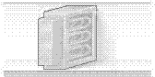

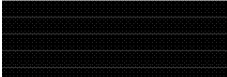





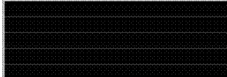

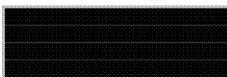
- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности нагрева.
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний calorifiera.





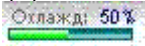
Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить calorifier	Логический	от ОРС	Сигнал на включение calorifiera.
	Общая авария	Логический	от ОРС	1 - на мнемосхеме появляется аварийное предупреждение ФБ.
	Переходный режим	Логический	от ОРС	Сигнал определения переходного режима работы calorifiera: запускается или останавливается.
	%Прогрева	Целый	от ОРС	Определяет мощность прогрева электроcalorifiera.
	Фиксация режима	Логический	от ОРС	Фиксирует режимы работы calorifiera: запуск или останов.
Динамические настройки ФБ	Включить ФБ	Логический	от ОРС	Информирует о текущем состоянии объекта. Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке. Состояние 0 (<i>выкл</i>) данного входа скрывает (делает невидимым) всякое представление электроcalorifiera. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно.
	ФБ используется	Логический	от ОРС	Определяет, используется ли электроcalorifier в данное время года.
	Параметр N	Вещественный	от ОРС	На этот вход подается значение соответствующего параметра, который отображается в виде графика в информационном окне ФБ. Число графиков не должно превышать трех. Число параметров определяется на странице свойств <i>Настройка</i> .

Выходы ФБ:

	Имя	Тип	Зависимости (формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.

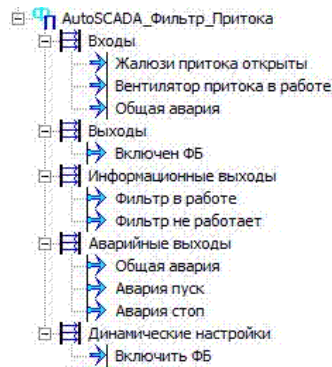
Информационные выходы	Имя	Тип	Зависимости (формулы, связи)	Примечание
	Калорифер не используется	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1, ФБ используется</i> = 0.	<p>Электронагреватель</p>  <p>Состояние: Отключен</p> <p>Электрический калорифер в данное время года не используется.</p>
	Калорифер запущен	Логический	Принимает значение 1 при динамических настройках <i>Включить ФБ = 1</i> (Во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет), <i>ФБ используется = 1</i> и отсутствии аварийных сигналов или режимов продувки или прогрева.	<p>Электронагреватель</p>  <p>Состояние: Запущен</p> 
	Калорифер остановлен	Логический	Принимает значение 1 при значении входа <i>ФБ используется = 1</i> или в режиме продувки или прогрева при отсутствии аварийных сигналов.	<p>Электронагреватель</p>  <p>Состояние: Остановлен</p> 
	Калорифер запускается	Логический	Принимает значение 1 при значении входа <i>ФБ используется = 1</i> в режиме прогрева.	<p>Электронагреватель</p> <p>Прогрев</p>  <p>Состояние: Запускается</p> <p>Предварительный разогрев ТЕНов для предотвращения термозудара при запуске вентилятора</p>
	Калорифер останавливается	Логический	Принимает значение 1 при значении входа <i>ФБ используется = 1</i> в режиме продувки.	<p>Электронагреватель</p> <p>Продувка</p>  <p>Состояние: Останавливается</p> <p>Охлаждение нагревательных элементов для предотвращения их перегрева при остановке вентилятора</p>
	Нагрев секция 1	Логический	Принимает значение 1 при значении входа <i>%Прогрева < 34%</i>	<p>Электронагреватель</p>  <p>Состояние: Запущен</p> 
	Нагрев секция 2	Логический	Принимает значение 1 при значении входа <i>%Прогрева от 34% до 66%</i> .	<p>Электронагреватель</p>  <p>Состояние: Запущен</p> 

Имя	Тип	Зависимости (формулы, связи)	Примечание
Нагрев секция 3	Логический	Принимает значение 1 при значении входа <i>%Прогрева > 66%</i>	<p>Электронагреватель</p>  <p>Нагрев: 67%</p> <p>Состояние: Запущен</p>
Режим продувка	Логический	Принимает значение 1 при значении входов <i>Включить калорифер = 0</i> и <i>Переходный режим = 1</i> .	<p>Электронагреватель</p> <p>Продувка</p>  <p>Состояние: Останавливается</p> <p>Охлаждение нагревательных элементов для предотвращения их перегрева при остановке вентилятора</p>
Режим прогрев	Логический	Принимает значение 1 при значении входов <i>Включить калорифер = 1</i> и <i>Переходный режим = 1</i> .	<p>Электронагреватель</p> <p>Прогрев</p>  <p>Состояние: Запускается</p> <p>Предварительный разогрев ТЕНов для предотвращения термоудара при запуске вентилятора</p>
Аварийные выходы	Общая авария	Логический	<p>Аварийный сигнал при условии отсутствия режимов продувки или прогрева калорифера.</p> <p>Электронагреватель</p> <p>Авария</p>  <p>Состояние: Авария</p> <p>Принят аварийный сигнал от термостата, контролирующего предельную температуру нагревательных элементов, управление электронагревателем отключено</p>
	Авария пуск	Логический	<p>Принимает значение 1 при комбинации входных сигналов <i>Общая авария = 1</i>, <i>Фиксация режима = 1</i> и отсутствии режима продувки.</p> <p>Отображение аварийного сигнала (см. «Общая авария»)</p>
	Авария стоп	Логический	<p>Принимает значение 1 при комбинации входных сигналов <i>Общая авария = 1</i>, <i>Фиксация режима = 0</i> и отсутствии режима продувки.</p> <p>Отображение аварийного сигнала(см. «Общая авария»)</p>
Выходы индикатора мощности	Включить (базовый)	Логический	<p><i>Включить охладитель = 1</i> (при отсутствии аварийных сигналов).</p> <p>Предназначен для отображения индикатора мощности на мнемосхеме и в информационном окне.</p>  <p>Охлажд.: 50%</p>
Нагрев	Целый		<p>Текущее значение процента мощности нагрева.</p>

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройка*.

Фильтр притока

Вид функционального блока в дереве объекта:



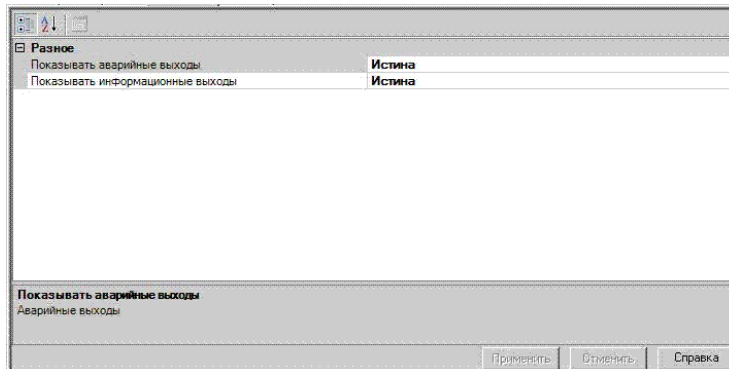
Визуальный функциональный блок *Фильтр притока* предназначен для определения состояния фильтра данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы фильтра и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон).

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы фильтра и аварий;
- Отдельное pop-up окно с изображением фильтра притока доступное только в режиме RT по нажатию на первое изображение. В данном ФБ отображает только анимацию работы фильтра и аварии.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние фильтра притока в каждый момент времени.

Страница свойств *Настройка* ФБ Фильтра притока выглядит следующим образом:



Элементы управления:

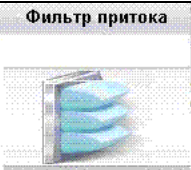
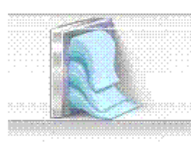


- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний фильтра.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Жалюзи притока открыты	Логический	от OPC	Определяет состояние жалюзи притока.
	Вентилятор притока в работе	Логический	от OPC	Определяет состояние вентилятора притока.
	Общая авария	Логический	от OPC	1 - на мнемосхеме появляется аварийное предупреждение ФБ.

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Динамические настройки ФБ	Включить ФБ	Логический	от ОРС	Значение «1» определяет, что этот элемент используется в конкретной вентиляционной установке. Начальное состояние «0» (выкл) данного входа скрывает (делает невидимым) всякое представление фильтра притока. Изменение состояния входа Включить ФБ меняет состояние выхода Включен ФБ соответственно

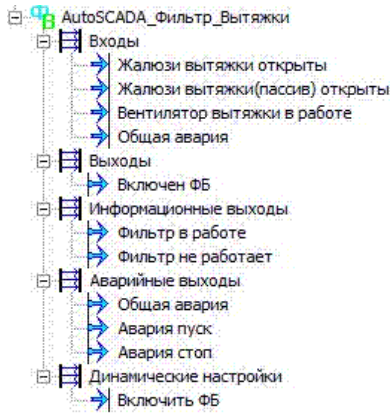
Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Лог.	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Фильтр в работе	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1</i> и <i>Жалюзи притока открыты = 1</i> и <i>Вентилятор притока в работе = 1</i> .	<p>Состояние: Нормальное</p>  <p>Состояние: Нормальное</p> 
	Фильтр не работает	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1</i> (Во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет) и <i>Жалюзи притока открыты = 0</i> или <i>Вентилятор притока в работе = 0</i> .	<p>Состояние: Нормальное</p>  <p>Состояние: Загрязнен</p> 
Аварийные выходы	Общая авария	Логический	Принимает значение 1 при значении входа <i>Общая авария = 1</i> .	Аварийный сигнал о загрязнении фильтра.
	Авария пуск	Логический	Равен 1, когда <i>Жалюзи притока открыты = 1</i> и <i>Вентилятор притока в работе = 1</i> и <i>Общая авария = 1</i>	Возникает при поступлении сигнала об общей аварии в процессе работы фильтра.
	Авария стоп	Логический	Равен 1, когда <i>Общая авария = 1</i> и (<i>Жалюзи притока открыты = 0</i> или <i>Вентилятор притока в работе = 0</i>).	Принимает значение 1 при поступлении сигнала об общей аварии при неработающем фильтре.

Наличие групп информационных и аварийных выходов настраивается на странице свойств ФБ *Настройки*.

Фильтр вытяжки

Вид функционального блока *Фильтр вытяжки* в дереве объекта:



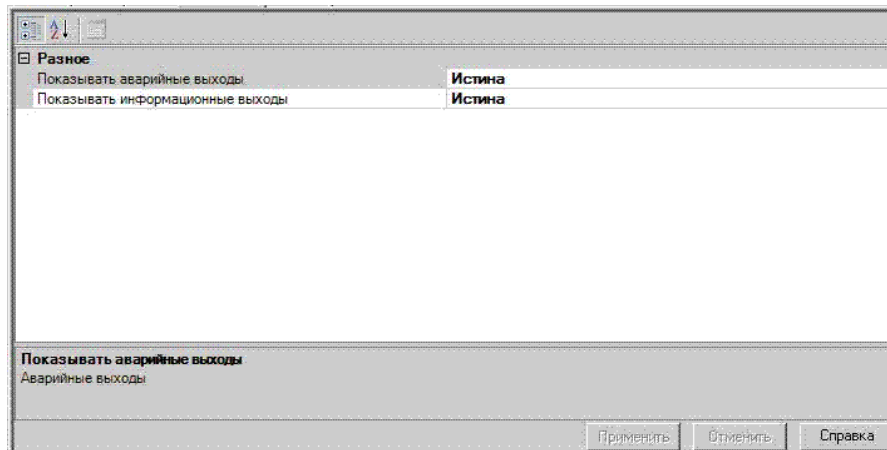
Визуальный функциональный блок *Фильтр вытяжки* предназначен для определения состояния фильтра данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы фильтра и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы фильтра и аварий;
- Отдельное pop-up окно с изображением фильтра вытяжки, доступное только в режиме RT по нажатию на первое изображение. В данном ФБ отображает только анимацию работы фильтра и аварии.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние фильтра вытяжки в каждый момент времени.

Страница свойств *Настройки* ФБ фильтр вытяжки выглядит следующим образом:



Элементы управления:


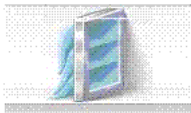


- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний фильтра.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Жалюзи вытяжки открыты	Логический	от OPC	Определяет состояние жалюзи вытяжки.
	Жалюзи вытяжки (пассив) открыты	Логический	от OPC	Определяет состояние жалюзи вытяжки.
	Вентилятор вытяжки в работе	Логический	от OPC	Определяет состояние вентилятора вытяжки.

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
	Общая авария	Логический	от ОРС	1 - на мнемосхеме появляется аварийное предупреждение ФБ.
Динамические настройки ФБ	Включить ФБ	Логический	от ОРС	Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке. Начальное состояние 0 (выкл.) данного входа скрывает (делает невидимым) всякое представление фильтра вытяжки. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответствующую

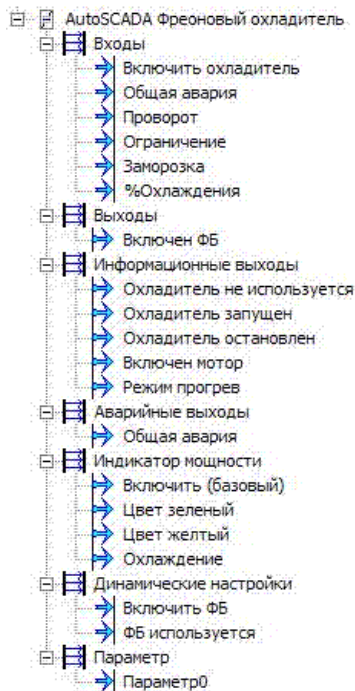
Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Фильтр в работе	Логический	Принимает значение 1 при значении входов <i>Включить ФБ</i> = 1 (Во всех последующих выходах в логике вход <i>Включить ФБ</i> = 1. В противном случае, ФБ на мнемосхеме отображаться не будет), <i>Жалюзи вытяжки открыты</i> = 1 или <i>Жалюзи вытяжки (пассив) открыты</i> = 1 и <i>Вентилятор вытяжки в работе</i> = 1.	 Состояние: Нормальное
	Фильтр не работает	Логический	Принимает значение 1 при значении входов <i>Включить ФБ</i> = 1 (Во всех последующих выходах в логике вход <i>Включить ФБ</i> = 1. В противном случае, ФБ на мнемосхеме отображаться не будет) и (<i>Жалюзи вытяжки открыты</i> = 0 или <i>Жалюзи вытяжки (пассив) открыты</i> = 0 или <i>Вентилятор вытяжки в работе</i> = 0).	 Состояние: Нормальное
Аварийные выходы	Общая авария	Логический	Принимает значение 1 при значении входа <i>Общая авария</i> = 1.	Аварийный сигнал о загрязнении фильтра.
	Авария пуск	Логический	Равен 1, когда (<i>Жалюзи вытяжки открыты</i> = 1 или <i>Жалюзи вытяжки (пассив) открыты</i> = 1) и <i>Вентилятор вытяжки в работе</i> = 1 и <i>Общая авария</i> = 1.	Возникает при поступлении сигнала об общей аварии в процессе работы фильтра.
	Авария стоп	Логический	Равен 1, когда <i>Общая авария</i> = 1 и (<i>Жалюзи вытяжки открыты</i> = 0 или <i>Жалюзи вытяжки (пассив) открыты</i> = 0) или <i>Вентилятор вытяжки в работе</i> = 0.	 Состояние: Загрязнен Возникает при поступлении сигнала об общей аварии при неработающем фильтре.
				 Состояние: Загрязнен

Наличие групп информационных и аварийных выходов настраивается на странице свойств ФБ *Настройки*.

Фреоновый охладитель

Вид функционального блока *Фреоновый охладитель* в дереве объекта:



Визуальный функциональный блок *Фреоновый охладитель* предназначен для определения состояния охладителя данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы охладителя и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы охладителя, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением фреонового охладителя, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы охладителя, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - зависимости от комбинации входных сигналов определяется и отображается состояние охладителя в каждый момент времени.

Страница свойств *Настройки* ФБ фреоновый охладитель выглядит следующим образом:

The screenshot shows the 'Настройки' (Settings) window for the 'Фреоновый охладитель' block. It contains a table with the following settings:

Разное	
Количество параметров для графика	1
Показывать аварийные выходы	Истина
Показывать выходы индикатора мощности	Истина
Показывать информационные выходы	Истина

Below the table, there is a section for 'Количество параметров для графика' (Number of parameters for the graph) with a sub-section for 'Параметры графика' (Graph parameters). At the bottom of the window are buttons for 'Применить' (Apply), 'Отменить' (Cancel), and 'Справка' (Help).

Элементы управления:

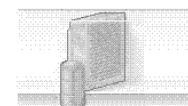
- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности охлаждения;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний охладителя.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить охладитель	Логический	от ОРС	1 - запускает охладитель.
	Общая авария	Логический	от ОРС	1 - на мнемосхеме появляется аварийное предупреждение ФБ.
	Проворот	Логический	от ОРС	Предназначен для отображения режима работы двигателя охладительного компрессора.
	Ограничение	Логический	от ОРС	Включение режима ограничения мощности охладителя.
Динамические настройки ФБ	Заморозка	Логический	от ОРС	Включает прогрев охладителя.
	%Охлаждения	Целый	от ОРС	Процент охлаждения фреонового охладителя.
Параметр	Включить ФБ	Логический	от ОРС	Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке. Начальное состояние 0 (выкл.) данного входа скрывает (делает невидимым) всякое представление фреонового охладителя. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно.
	ФБ используется	Логический	от ОРС	Определяет, используется ли охладитель в данное время года.
Параметр	ПараметрN	Вещественный	от ОРС	На этот вход подается значение того параметра, который отображается на графике в информационном окне ФБ. Число графиков не должно превышать трех. Число параметров определяется на странице свойств <i>Настройки</i> .



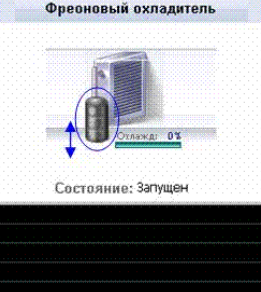

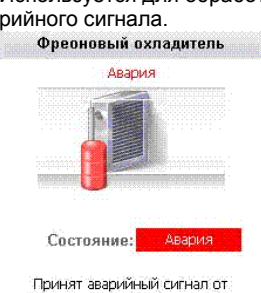
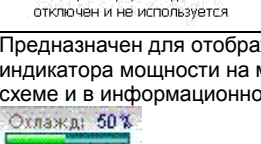
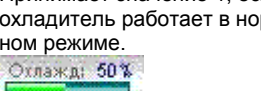
Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Охладитель не используется	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1</i> , <i>ФБ используется = 0</i> и отсутствии аварийных сигналов.	Фреоновый охладитель



Состояние: Отключен

Фреоновый охладитель в данное время года не используется.

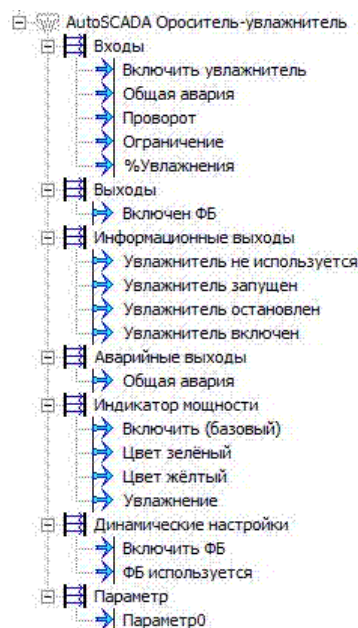
Имя	Тип	Зависимости (Формулы, связи)	Примечание
Охладитель запущен	Логический	Принимает значение 1 при динамических настройках <i>Включить ФБ = 1</i> (Во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет), <i>ФБ используется = 1</i> и при значении входа <i>Включить охладитель = 1</i> и отсутствии аварийных сигналов.	<p>Фреоновый охладитель</p>  <p>Состояние: Запущен</p>
Охладитель остановлен	Логический	Принимает значение 1 при значении входов <i>ФБ используется = 1</i> И <i>Включить охладитель = 0</i> и отсутствии аварийных сигналов.	<p>Фреоновый охладитель</p>  <p>Состояние: Остановлен</p>
Включен мотор	Логический	Принимает значение 1, когда вход <i>Проворот = 1</i> .	<p>Фреоновый охладитель</p>  <p>Состояние: Запущен</p>
Режим прогрева	Логический	Когда вход <i>Заморозка = 1</i> И выход <i>Общая авария = 0</i> , а также вход группы <i>Динамические настройки ФБ используется = 1</i> .	<p>Фреоновый охладитель</p> <p>Прогрев</p>  <p>Состояние: Прогрев</p> <p>Принят сигнал об опасности обмерзания радиатора испарителя, работа охлаждителя временно приостановлена</p>
Аварийные выходы	Общая авария	Логический	<p><i>Общая авария = 1</i>.</p> <p>Используется для обработки аварийного сигнала.</p> <p>Фреоновый охладитель</p> <p>Авария</p>  <p>Состояние: Авария</p> <p>Принят аварийный сигнал от термореле двигателя компрессора, охладитель отключен и не используется</p>
Выходы индикатора мощности	Включить (базовый)	Логический	<p><i>Включить охладитель = 1</i> (при отсутствии аварийных сигналов).</p> <p>Предназначен для отображения индикатора мощности на мнемосхеме и в информационном окне.</p> 
	Цвет зеленый	Логический	<p>Выход группы <i>Индикатор мощности Включить (базовый) = 1</i> И вход <i>Ограничение = 0</i></p> <p>Принимает значение 1, если охладитель работает в нормальном режиме.</p> 
	Цвет желтый	Логический	<p>Выход группы <i>Индикатор мощности Включить (ба-</i></p> <p>Принимает значение 1, если охладитель работает в режиме</p>

Имя	Тип	Зависимости (Формулы, связи)	Примечание
Охлаждение	Целый	$з\text{овый}} = 1 \text{ И вход } \text{Ограничение} = 1$	ограничения. Охлажд: 47% Текущее значение процента мощности охлаждения.

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройки*.

Ороситель-увлажнитель

Вид функционального блока *Ороситель-увлажнитель* в дереве *Объекта*:



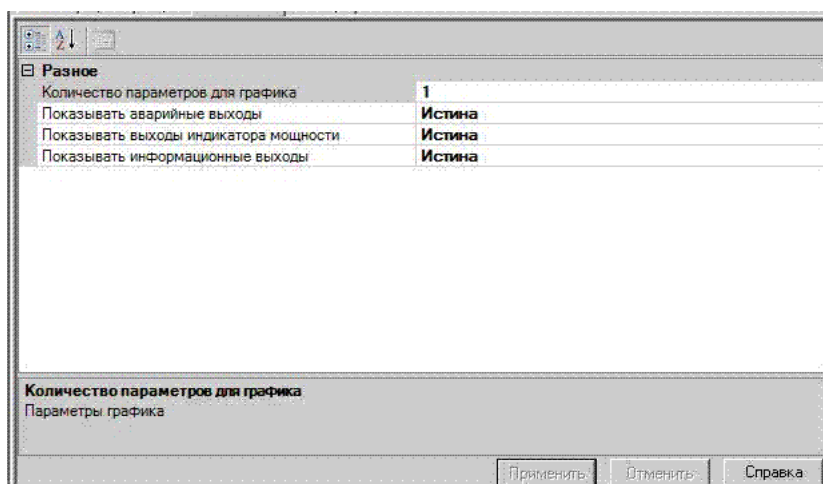
Визуальный функциональный блок *Ороситель-увлажнитель* предназначен для определения состояния увлажнителя данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы увлажнителя и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон).

- Первый – изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся «перетаскиванием» ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы увлажнителя, аварий и параметра мощности в виде прогресс бара;
- Второй – отдельное «pop-up» окно с изображением оросителя-увлажнителя, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы увлажнителя, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние увлажнителя в каждый момент времени.

Страница свойств *Настройки* ФБ ороситель-увлажнитель выглядит следующим образом:



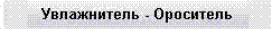
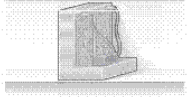
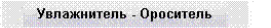
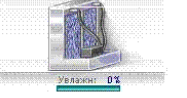
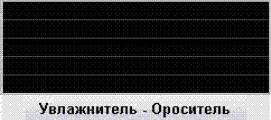

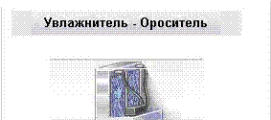
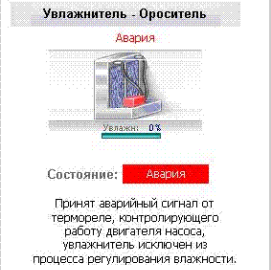
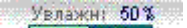


Элементы управления:

- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности увлажнения;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний увлажнителя.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить увлажнитель	Лог.	от OPC	Команда. При значении <i>Истина</i> запускает ороситель-увлажнитель.
	Общая авария	Лог.	от OPC	Команда. Если <i>Истина</i> , то на мнемосхеме появляется аварийное предупреждение этого ФБ.
	Проворот	Лог.	от OPC	Команда предназначена для определения режима <i>проворот</i> работы увлажнителя.
	Ограничение	Лог.	от OPC	Определяет наличие режима ограничения мощности увлажнителя.
	%Увлажнения	Целый	от OPC	Отображает мощность увлажнения оросителя-увлажнителя.
Динамические настройки ФБ	Включить ФБ	Лог.	от OPC	Значение <i>1</i> определяет, будет ли этот элемент использоваться в конкретной вентиляционной установке. Состояние <i>0 (выкл.)</i> данного входа скрывает (делает невидимым) всякое представление оросителя-увлажнителя. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно. Информирование о текущем состоянии объекта.
	ФБ используется	Лог.	от OPC	
Параметр	Параметр N	Вещ.		На этот вход подается значение того параметра, который должен будет отображаться на графике в информационном окне ФБ. Число параметров определяется на странице свойств <i>Настройки</i> . Число графиков не должно превышать трех.

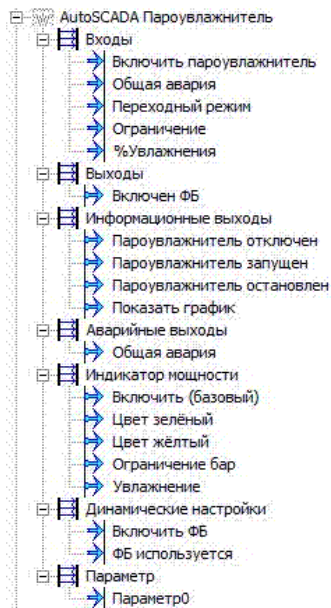
Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы Информационные выходы	Включен ФБ	Лог.	<i>Включить ФБ = 1</i>	Информирует о состоянии ФБ.
	Увлажнитель не используется	Лог.	<i>Включить ФБ = 1, ФБ используется = 0</i>	Определяет, используется ли увлажнитель в данное время года. 
	Увлажнитель запущен	Лог.	<i>Включить ФБ = 1 (Во всех последующих выходах в логике вход Включить ФБ = 1. В противном случае, ФБ на мнемосхеме отображаться не будет), ФБ используется = 1, Включить увлажнитель = 1</i>	 Состояние: Отключен Ороситель в данное время года не используется. 
	Увлажнитель остановлен	Лог.	<i>ФБ используется = 1, Включить увлажнитель = 0</i>	 Состояние: Запущен 
	Увлажнитель включен	Лог.	<i>Проворот = 1, Включить увлажнитель = 1.</i>	 Состояние: Остановлен 
Аварийные выходы	Общая авария	Лог.	<i>Общая авария = 1</i>	 Состояние: авария Принят аварийный сигнал от термореле, контролирующего работу двигателя насоса, увлажнитель исключен из процесса регулирования влажности.
Выходы индикатора мощности	Включить (базовый)	Лог.	<i>Включить увлажнитель = 1.</i>	Выход предназначен для отображения индикатора мощности на мнемосхеме.   
	Цвет зеленый	Лог.	<i>Включить (базовый) = 1, Ограничение = 0.</i>	
	Цвет желтый	Лог.	<i>Включить (базовый) = 1, Ограничение = 1</i>	
	Увлажнение	Целый		Текущее значение процента увлажнения.

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройка*.

Пароувлажнитель

Вид функционального блока *Пароувлажнитель* в дереве объекта:



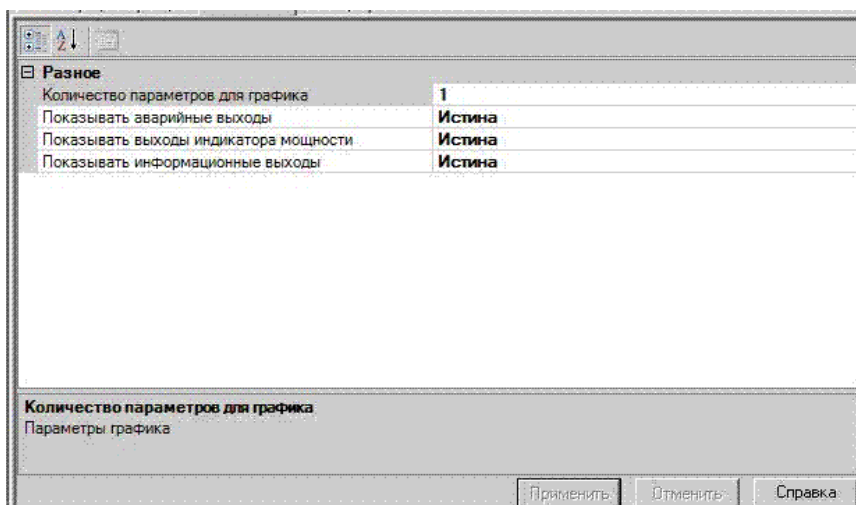
Визуальный функциональный блок *Пароувлажнитель* предназначен для определения состояния увлажнителя данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы увлажнителя и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон).

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы увлажнителя, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением пароувлажнителя, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы увлажнителя, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние увлажнителя в каждый момент времени.

Страница свойств *Настройки* ФБ пароувлажнитель выглядит следующим образом:



Элементы управления:

- **Количество параметров для графика** – определяет количество отображаемых графиков на тренде в информационном окне ФБ;

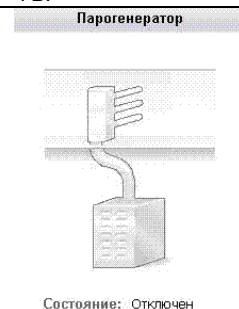
- **Показывать аварийные выходы** – определяет наличие выходов, относящихся к авариям;
- **Показывать выходы индикатора мощности** – определяет наличие выходов, относящихся к мощности увлажнения;
- **Показывать информационные выходы** – определяет наличие выходов, относящихся к отображению состояний увлажнителя.

Входы ФБ:

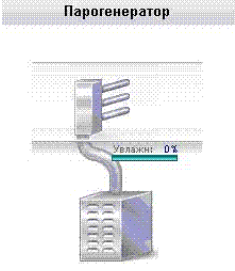
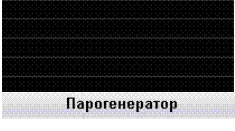
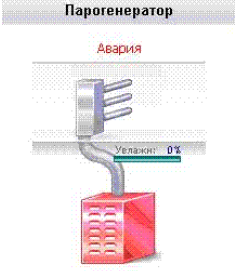
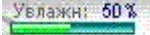
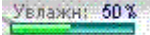

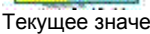
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить пароувлажнитель	Логический	от ОРС	1 - запускает пароувлажнитель.
	Общая авария	Логический	от ОРС	1 - на мнемосхеме появляется аварийное предупреждение ФБ.
	Переходный режим	Логический	от ОРС	Сигнал для определения переходного режима пароувлажнителя. Отключает Информационный выход <i>Пароувлажнитель запущен</i> .
	Ограничение	Логический	от ОРС	Определяет наличие режима ограничения мощности увлажнителя.
	%Увлажнение	Целый	от ОРС	Определяет мощность увлажнения пароувлажнителя.
Динамические настройки ФБ	Включить ФБ	Логический	от ОРС	Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке. Начальное состояние 0 (<i>выкл.</i>) данного входа скрывает (делает невидимым) всякое представление пароувлажнителя. Изменение состояния входа Включить ФБ меняет состояние выхода Включен ФБ соответственно.
	ФБ используется	Логический	от ОРС	Определяет, используется ли пароувлажнитель в данное время года.
Параметр	ПараметрN	Вещественный	от ОРС	На этот вход подается значение соответствующего параметра, который отображается в виде графика в информационном окне ФБ. Число графиков не должно превышать трех. Число параметров определяется на странице свойств <i>Настройки</i> .

Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Увлажнитель отключен	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1, ФБ используется</i> = 0 и отсутствии аварийных сигналов.	



Пароувлажнитель в данное время года не используется.

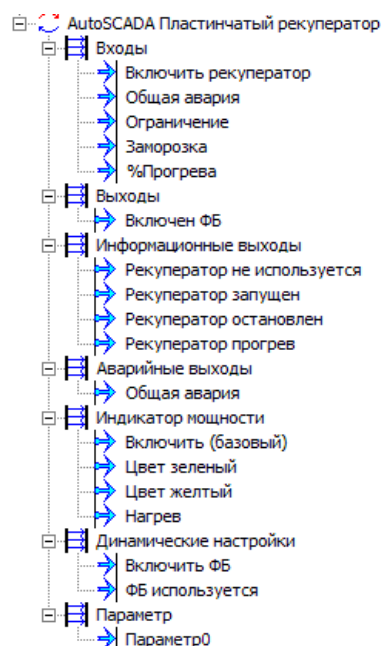
Имя	Тип	Зависимости (Формулы, связи)	Примечание
Увлажнитель запущен	Логический	Принимает значение 1 при динамических настройках <i>Включить ФБ = 1</i> (Во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет), <i>ФБ используется = 1</i> , при значении входа <i>Включить паровлажнитель = 1</i> и отсутствии аварии и сигналов о переходном процессе.	 <p>Парогенератор</p> <p>Состояние: Запущен</p>
Увлажнитель остановлен	Логический	Принимает значение 1 при значении входов <i>ФБ используется = 1</i> И <i>Включить паровлажнитель = 0</i> и отсутствии сигналов о переходном процессе и аварии.	 <p>Парогенератор</p> <p>Состояние: Остановлен</p>
Аварийные выходы	Общая авария	Логический	Используется для обработки аварийного сигнала.  <p>Парогенератор</p> <p>Состояние: Авария</p> <p>Принят аварийный сигнал от блока управления увлажнителем, увлажнитель исключен из процесса регулирования влажности.</p>
Выходы индикатора мощности	Включить (базовый)	Логический	Включается автоматически при значении входа <i>Включить паровлажнитель = 1</i> . 
	Цвет зеленый	Логический	Выход группы Индикатор мощности <i>Включить (базовый) = 1</i> И вход <i>Ограничение = 0</i> . 
	Цвет желтый	Логический	Выход группы Индикатор мощности <i>Включить (базовый) = 1</i> И вход <i>Ограничение = 1</i> . 
	Увлажнение	Целый	Текущее значение процента увлажнения. 

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройки*.

- ❖ **Важно!** ФБ из библиотеки *вентиляция*, возможно использовать только при наличии исполнительной системы PRO. Паровлажнитель и парогенератор - идентичное оборудование в данном контексте.

Рекуператор пластинчатый

Вид функционального блока *Рекуператор пластинчатый* в дереве объекта:



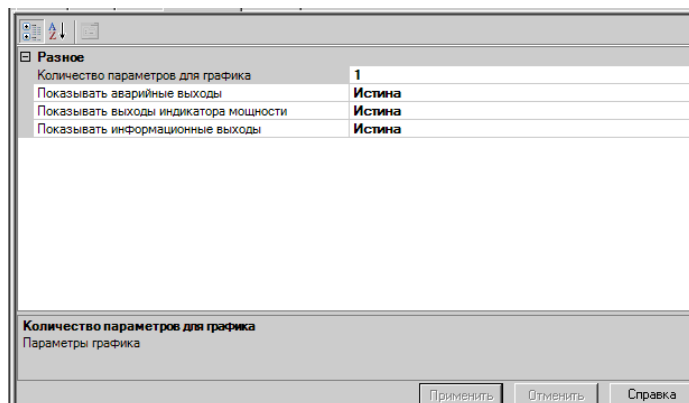
Визуальный функциональный блок *Пластинчатый рекуператор* предназначен для определения состояния рекуператора данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы рекуператора и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон).

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы рекуператора, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением роторного рекуператора, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы рекуператора, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - В зависимости от комбинации входных сигналов определяется и отображается состояние рекуператора в каждый момент времени.

Страница свойств *Настройки* ФБ пластинчатый рекуператор выглядит следующим образом:



Элементы управления:


- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;



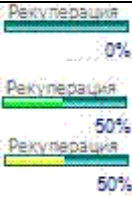
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности прогрева
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний рекуператора.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить рекуператор	Лог.	от ОРС	Команда. При значении 1 запускает рекуператор.
	Общая авария	Лог.	от ОРС	Команда. Если 1, то на мнемосхеме появляется аварийное предупреждение этого ФБ.
	Ограничение	Лог.	от ОРС	Команда, используемая для индикации режима ограничения мощности рекуператора.
	Заморозка	Лог.	от ОРС	Команда, используемая для определения возможного обмерзания рекуператора.
Динамические настройки ФБ	%Прогрева	Цел.	от ОРС	Отображает мощность прогрева пластинчатого рекуператора.
	Включить ФБ	Лог.	от ОРС	Значение 1 определяет, будет ли этот элемент использоваться в конкретной вентиляционной установке.
Параметр	ФБ используется	Лог.	от ОРС	Информирует о текущем состоянии объекта. Определяет, используется ли в данное время пластинчатый рекуператор.
	Параметр N	Вещ.		На этот вход подается значение того параметра, который должен будет отображаться на графике в информационном окне ФБ. Число параметров определяется на странице свойств <i>Настройки</i> . Число графиков не превышает трех.

Выходы ФБ:

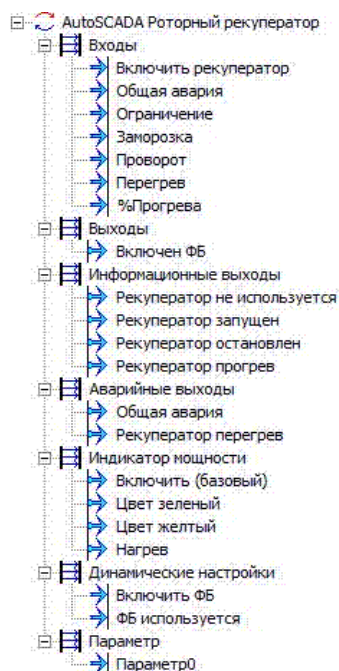
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы Информационные выходы	Включен ФБ	Лог.	$Включить\ ФБ = 1$	
	Рекуператор не используется	Лог.	$Включить\ ФБ = 1$ и $ФБ\ используется = 0$.	
	Рекуператор запущен	Лог.	$ФБ\ используется = 1$, $Включить\ рекуператор = 1$	
	Рекуператор остановлен	Лог.	$ФБ\ используется = 1$, $Включить\ рекуператор = 0$	

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
	Рекуператор прогрев	Лог.	$Заморозка = 1$	 <p>Состояние: Прогрев</p> <p>Обнаружено сочетание условий, при которых возможно переобращение рабочего тела, работа рекуператора временно приостановлена.</p>
Аварийные выходы	Общая авария	Лог.	$Общая авария = 1.$	 <p>Состояние: Авария</p> <p>Принят сигнал о неисправности привода перепускной заслонки, рекуператор отключен и не используется.</p>
Индикатор мощности	Включить (базовый)	Лог.	$ФБ\ используется = 1$ $Включить\ рекуператор = 1$	 <p>Отображается текущее значение мощности нагрева</p>
	Цвет зеленый	Лог.	$Включить\ (базовый) = 1,$ $Ограничение = 0.$	
	Цвет желтый	Лог.	$Включить\ (базовый) = 1,$ $Ограничение = 1$	
	Нагрев	Лог.	$\%Прогрева$	

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройки*.

Рекуператор роторный

Вид функционального блока *Рекуператор роторный* в дереве объекта:



Визуальный функциональный блок *Роторный рекуператор* предназначен для определения состояния рекуператора данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к

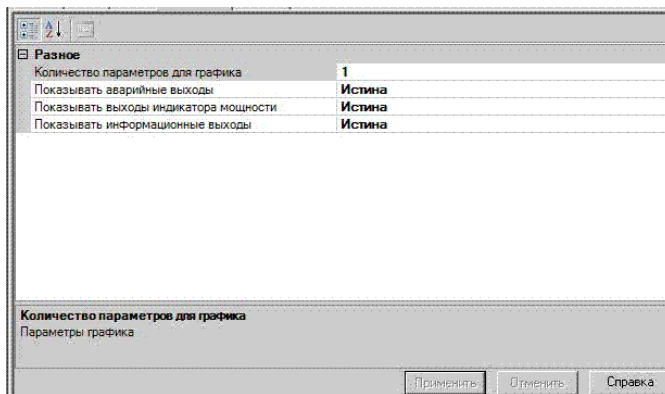
этому ФБ. ФБ предназначен для визуализации технологических параметров работы рекуператора и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы рекуператора, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением роторного рекуператора, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы рекуператора, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая – в зависимости от комбинации входных сигналов определяется и отображается состояние рекуператора в каждый момент времени.

Страница свойств *Настройки* ФБ роторный рекуператор выглядит следующим образом:



Элементы управления:

- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности прогрева;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний рекуператора.

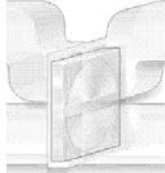
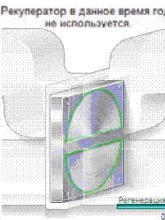
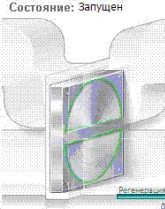
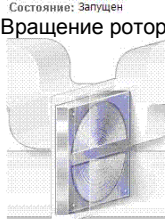

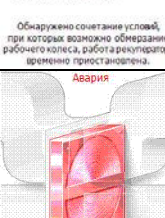

Входы ФБ:

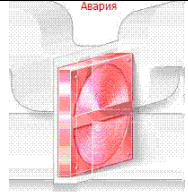

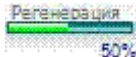
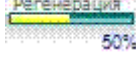
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить рекуператор	Лог.	от OPC	Команда. При значении 1 запускает рекуператор.
	Общая авария	Лог.	от OPC	Команда. Если 1, то на мнемосхеме появляется аварийное предупреждение этого ФБ.
	Ограничение	Лог.	от OPC	Команда, используемая для индикации режима ограничения мощности рекуператора.
	Заморозка	Лог.	от OPC	Команда, используемая для определения возможного обмерзания рекуператора.
	Проворот	Лог.	от OPC	Команда, соответствующая режиму мягкого запуска.
	Перегрев	Лог.	от OPC	Команда, принимающая значение 1 в случае аварийного перегрева рекуператора
	%Прогрева	Цел.	от OPC	Отображает мощность прогрева роторного рекуператора.
Динамические настройки ФБ	Включить ФБ	Лог.	от OPC	Значение 1 определяет, будет ли этот элемент использоваться в конкретной вентиляционной установке.
	ФБ используется	Лог.	от OPC	Информирует о текущем состоянии объекта. Определяет, используется ли в данное время роторный реку-

ператор.

Параметр ПараметрN Вещ. На этот вход подается значение того параметра, который должен будет отображаться на графике в информационном окне ФБ. Число параметров определяется на странице свойств *Настройки*. Число графиков не превышает трех.

Выходы ФБ:

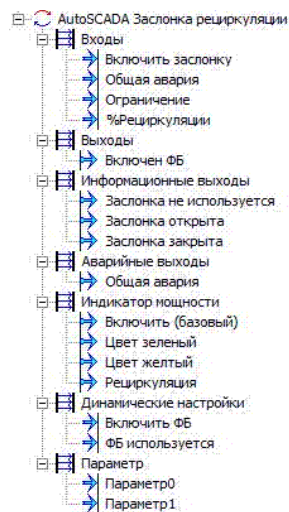
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы Информационные выходы	Включен ФБ	Лог.	<i>Включить ФБ = 1.</i>	 Состояние: Отключен
	Рекуператор не используется	Лог.	<i>Включить ФБ = 1 и ФБ используется = 0.</i>	 Рекуператор в данное время года не используется
	Рекуператор включен	Лог.	<i>Включить ФБ = 1</i> (во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет), <i>ФБ используется = 1, Включить рекуператор = 1.</i>	 Состояние: Запущен
	Рекуператор запущен	Лог.	<i>Проворот = 1</i> или <i>%Прогрева > 0</i>	 Состояние: Запущен
	Рекуператор остановлен	Лог.	<i>ФБ используется = 1, Включить рекуператор = 1.</i>	 Состояние: Остановлен
Рекуператор прогрев	Лог.	<i>Заморозка = 1.</i>	 Состояние: Прогрев	
Аварийные выходы	Общая авария	Лог.	<i>Общая авария = 1.</i>	 Состояние: Авария Мигает надпись Авария

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
	Рекуператор перегрев	Лог.	<i>Перегрев = 1.</i>	 <p>Состояние: Авария</p> <p>Принят сигнал о неисправности привода рабочего колеса, рекуператор отключен и не используется.</p> <p>Мигает надпись Авария</p>
Индикатор мощности	Включить (базовый)	Лог.	<i>ФБ используется = 1 и Включить рекуператор = 1 или Проворот = 1 или Ограничение = 1.</i>	
	Цвет зеленый	Лог.	<i>Включить (базовый) = 1, Ограничение = 0.</i>	
	Цвет желтый	Лог.	<i>Включить (базовый) = 1, Ограничение = 1.</i>	
	Нагрев	Лог.	<i>%Прогрева.</i>	Отображается текущее значение мощности нагрева

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройки*.

Заслонка рециркуляции

Вид функционального блока *Заслонка рециркуляции* в дереве объекта:



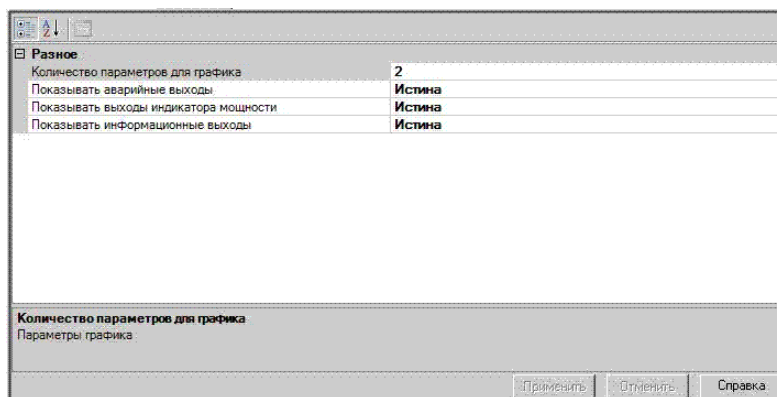
Визуальный функциональный блок *Заслонка рециркуляции* предназначен для определения состояния данного объекта. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы заслонки и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы заслонки, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением заслонки рециркуляции, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы заслонки, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние заслонки в каждый момент времени.

Страница свойств *Настройки* ФБ заслонки рециркуляции выглядит следующим образом:




Элементы управления:


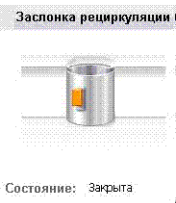

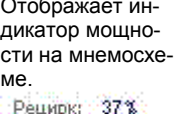
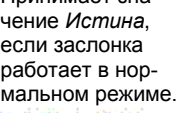
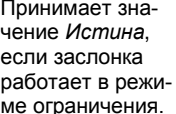
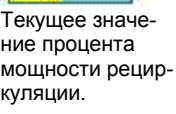
- **Количество параметров для графика** – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- **Показывать аварийные выходы** – определяет наличие выходов, относящихся к авариям;
- **Показывать выходы индикатора мощности** – определяет наличие выходов, относящихся к мощности рециркуляции;
- **Показывать информационные выходы** – определяет наличие выходов, относящихся к отображению состояний заслонки.

Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить заслонку	Логический	от OPC	Сигнал включения заслонки
	Общая авария	Логический	от OPC	Аварийный сигнал заслонки рециркуляции
	Ограничение	Логический	от OPC	Определяет наличие режима ограничения мощности рециркуляции.
	%Рециркуляции	Целый	от OPC	Определяет процент рециркуляции заслонки.
Динамические настройки ФБ	Включить ФБ	Логический	от OPC	Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке. Начальное состояние 0 (выкл.) данного входа скрывает (делает невидимым) всякое представление заслонки. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно.
	ФБ используется	Логический	от OPC	Определяет, используется ли заслонка в данное время года.
Параметр	ПараметрN	Вещественный	от OPC	На этот вход подается значение того параметра, который отображается в виде графика в информационном окне ФБ. Число графиков не должно превышать трех. Число параметров определяется на странице свойств <i>Настройки</i> .

Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Заслонка не используется	Логический	Принимает значение 1 при значении динамических настроек <i>Включить ФБ</i> = 1, <i>ФБ используется</i> = 0 и отсутствии аварийных сигналов.	 Состояние: Отсутствует Сезонный клапан в данное время года не используется.

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
	Заслонка открыта	Логический	Принимает значение 1 при динамических настройках <i>Включить ФБ = 1</i> (во всех последующих выходах в логике вход <i>Включить ФБ</i> = 1. В противном случае, ФБ на мнемосхеме отображаться не будет), <i>ФБ используется = 1</i> , значении входа <i>Включить заслонку = 1</i> , если отсутствуют аварийные сигналы.	 <p>Состояние: Открыта</p>
	Заслонка закрыта	Логический	Принимает значение 1 при значении входов <i>ФБ используется = 1</i> и <i>Включить заслонку = 0</i> .	 <p>Состояние: Закрыта</p>
Аварийные Выходы	Общая авария	Логический	Принимает значение 1 при значении входа <i>Общая авария = 1</i> .	Используется для обработки аварийного сигнала.  <p>Состояние: Авария</p>
Выходы индикатора мощности	Включить (базовый)	Логический	Принимает значение 1 при значении входов <i>Включить заслонку = 1</i> и отсутствии аварийных сигналов.	Отображает индикатор мощности на мнемосхеме. 
	Цвет зеленый	Логический	Выход группы <i>Индикатор мощности Включить (базовый) = 1</i> И вход <i>Ограничение = 0</i> .	Принимает значение <i>Истина</i> , если заслонка работает в нормальном режиме. 
	Цвет желтый	Логический	Выход группы <i>Индикатор мощности Включить (базовый) = 1</i> И вход <i>Ограничение = 1</i> .	Принимает значение <i>Истина</i> , если заслонка работает в режиме ограничения. 
	Рециркуляция	Целый		Текущее значение процента мощности рециркуляции. 

Наличие групп информационных, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройка*.

Водяной охладитель

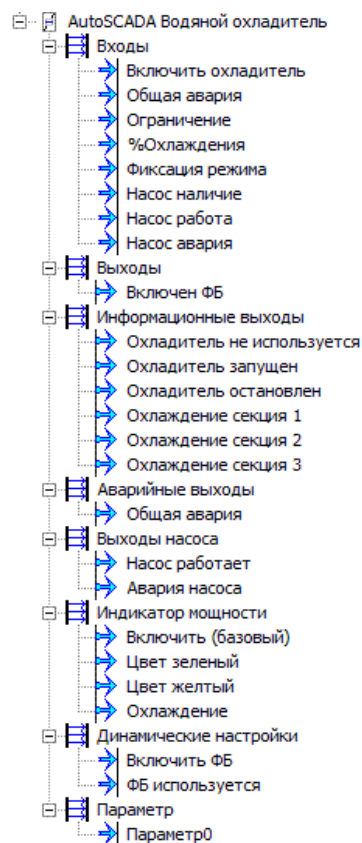
Визуальный функциональный блок *Водяной охладитель* предназначен для определения состояния охладителя данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы охладителя и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы охладителя, аварий и параметра мощности в виде прогресс бара;
- Отдельное поп-уп окно с изображением водяного охладителя, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный

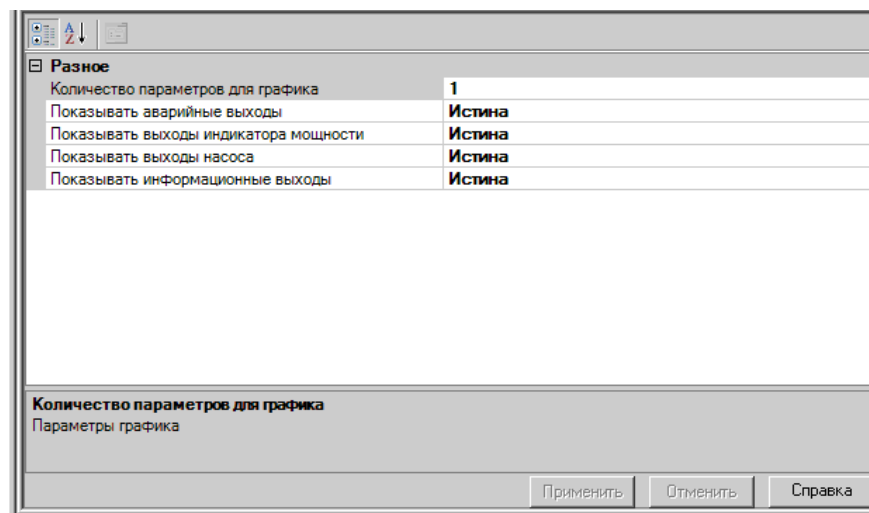
набор – анимация работы охладителя, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Вид функционального блока *Водяной охладитель* в дереве *Объекта*:



Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние охладителя в каждый момент времени.

Страница свойств *Настройки* ФБ водяной охладитель выглядит следующим образом:



Элементы управления:


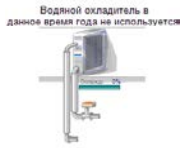
- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности охлаждения;
- *Показывать выходы насоса* – определяет наличие выходов, относящихся к состоянию насоса охладителя;


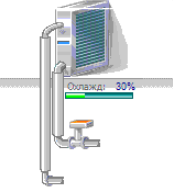
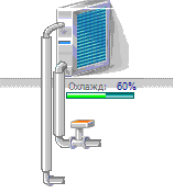
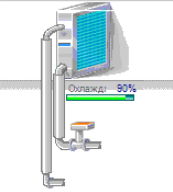

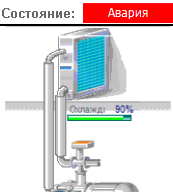
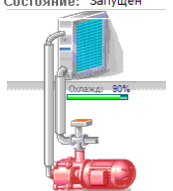
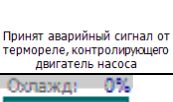
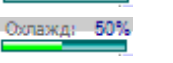
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний охладителя.

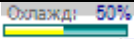
Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить охладитель	Лог.	от OPC	Команда. При значении 1 запускает охладитель.
	Общая авария	Лог.	от OPC	Команда. Если 1, то на мнемосхеме появляется аварийное предупреждение этого ФБ.
	Ограничение	Лог.	от OPC	Команда, используемая для индикации режима ограничения мощности охладителя.
	%Охлаждения	Целый	от OPC	Отображает процент охлаждения водяного охладителя.
	Фиксация режима	Лог.	от OPC	Команда предназначена для определения режимов работы калорифера: запуска или останова.
	Насос наличие	Лог.	от OPC	Определяет есть ли у данного охладителя насос.
	Насос работа	Лог.	от OPC	Команда, определяющая режим работы насоса: запуск или останов.
Динамические настройки ФБ	Насос авария	Лог.	от OPC	Команда, соответствующая аварийному сигналу от насоса охладителя.
	Включить ФБ	Лог.	от OPC	Информирует о текущем состоянии объекта. Значение 1 определяет, будет ли этот элемент использоваться в конкретной вентиляционной установке.
Параметр	ФБ используется	Лог.	от OPC	Определяет используется ли охладитель в данное время года.
	ПараметрN	Вещ.		На этот вход подается значение того параметра, который должен будет отображаться на графике в информационном окне ФБ. Число параметров определяется на странице свойств <i>Настройки</i> . Число графиков не должно превышать трех.

Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Лог.	$Включить\ ФБ = 1.$	
Информационные выходы	Охладитель не используется	Лог.	$Включить\ ФБ = 1, ФБ\ используется = 0.$	 Состояние: Отключен
	Охладитель запущен	Лог.	$Включить\ ФБ = 1$ (Во всех последующих выходах в логике вход $Включить\ ФБ = 1$. В противном случае, ФБ на мнемосхеме отображаться не будет), $ФБ\ используется = 1$, $Включить\ охладитель = 1.$	 Состояние: Запущен

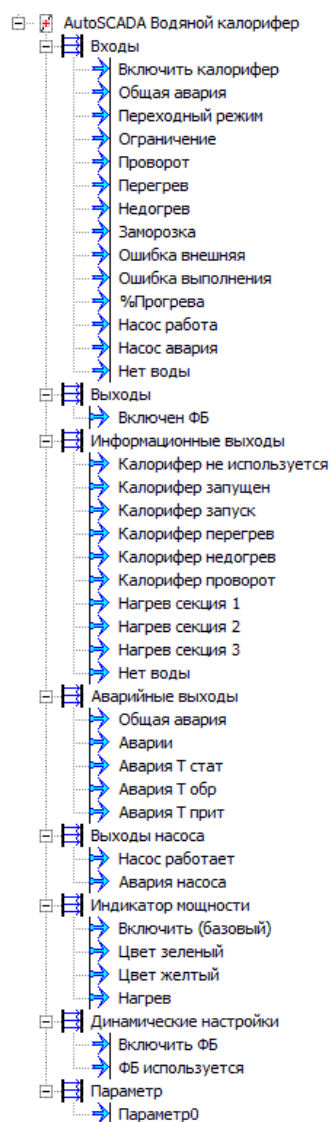
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
	Охладитель остановлен	Лог.	$ФБ\ используется = 1, Включить\ охладитель = 0.$	
	Охлаждение секция 1	Лог.	Принимает значение 1 при значении входа %Охлаждения < 33%.	Состояние: Остановлен 
	Охлаждение секция 2	Лог.	Принимает значение 1 при значении входа %Охлаждения от 33% до 66%.	Состояние: Запущен 
	Охлаждение секция 3	Лог.	Принимает значение 1 при значении входа %Охлаждения > 66%.	Состояние: Запущен 
Аварийные выходы	Общая авария	Лог.	$Общая\ авария = 1.$	Состояние: Запущен Мигает надпись Авария. 
Выходы насоса	Насос работает	Лог.	$Насос\ наличие = 1, Насос\ работа = 1.$	Состояние: Авария 
	Авария насоса	Лог.	$Насос\ наличие = 1, Насос\ авария = 1.$	Состояние: Запущен 
				Состояние: Запущен Принят аварийный сигнал от термореле, контролирующего двигатель насоса
Выходы индикатора мощности	Включить (базовый)	Лог.	$Включить\ охладитель = 1.$	
	Цвет зеленый	Лог.	$Включить\ охладитель = 1, Ограничение = 0.$	

Имя	Тип	Зависимости (Формулы, связи)	Примечание
Цвет желтый	Лог.	$Включить\ охладитель = 1,$ $Ограничение = 1.$	
Охлаждение	Целый	$\%Охлаждения.$	Текущее значение процента мощности охлаждения.

Наличие групп информационных, насосов, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ *Настройки*.

Водяной калорифер

Вид функционального блока *Водяной калорифер* в дереве объекта:



Визуальный функциональный блок *Водяной калорифер* предназначен для определения состояния калорифера данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы калорифера и не предусматривает функций управления.

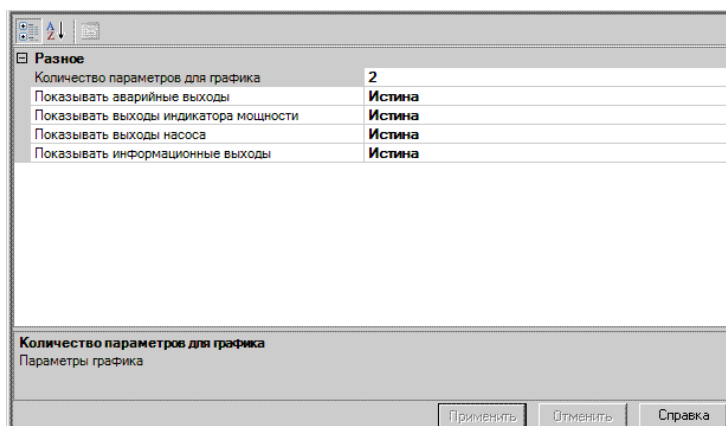
ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы калорифера, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением водяного калорифера, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный

набор – анимация работы калорифера, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние калорифера в каждый момент времени.

Страница свойств *Настройки* ФБ водяной калорифер выглядит следующим образом:



Элементы управления:

- *Количество параметров для графика* – определяет количество отображаемых графиков на тренде в информационном окне ФБ;
- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать выходы индикатора мощности* – определяет наличие выходов, относящихся к мощности нагрева;
- *Показывать выходы насоса* – определяет наличие выходов, относящихся к состоянию насоса калорифера;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний калорифера.



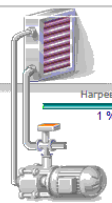
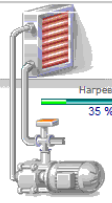
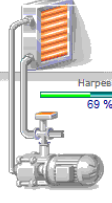
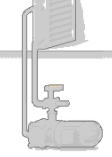
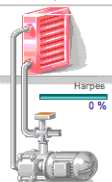
Входы ФБ:

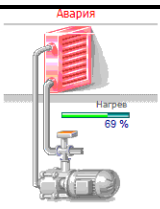
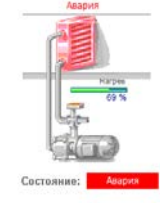


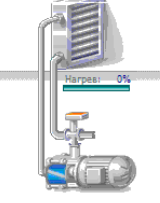
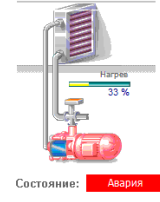
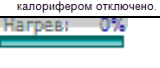
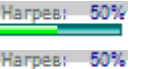
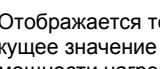
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Включить калорифер	Лог.	от OPC	Команда. При значении 1 запускает калорифер.
	Общая авария	Лог.	от OPC	Команда. Если 1, то на мнемосхеме появляется аварийное предупреждение этого ФБ.
	Переходный режим	Лог.	от OPC	Команда предназначена для определения режимов работы калорифера: запускается или останавливается.
	Ограничение	Лог.	от OPC	Команда, используемая для индикации режима ограничения мощности калорифера.
	Проворот	Лог.	от OPC	Команда, соответствующая режиму мягкого запуска.
	Перегрев	Лог.	от OPC	Команда, принимающая значение 1 в случае, когда ограничение температуры обратной воды не выше установленного предела.
	Недогрев	Лог.	от OPC	Команда, принимающая значение 1 в случае, когда ограничение температуры обратной воды не ниже установленного предела.
	Заморозка	Лог.	от OPC	Команда, используемая для определения аварийной ситуации в случае обмерзания калорифера и аварийного понижения температуры воды на выходе калорифера.
	Ошибка внешняя	Лог.	от OPC	Команда – аварийный сигнал от датчика температуры. Используется в определении аварийной ситуации.
	Ошибка выполнения	Лог.	от OPC	Команда – аварийный сигнал от термостата за калорифером. Используется в определении аварийной ситуации.
	%Прогрева	Цел.	от OPC	Отображает мощность прогрева водяного

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Динамические настройки ФБ	Насос Работа	Лог.	от ОРС	калорифера. Команда предназначена для определения режима работы насоса калорифера: запуска или останова.
	Насос Авария	Лог.	от ОРС	Команда, соответствующая аварийному сигналу от насоса калорифера.
	Нет воды	Лог.	от ОРС	Команда, определяющая готовность калорифера к запуску.
	Включить ФБ	Лог.	от ОРС	Значение 1 определяет, будет ли этот элемент использоваться в конкретной вентиляционной установке.
	ФБ используется	Лог.	от ОРС	Информирует о текущем состоянии объекта. Определяет, используется ли в данное время водяной калорифер.
Параметр	ПараметрN	Вещ.		На этот вход подается значение того параметра, который должен будет отображаться на графике в информационном окне ФБ. Число параметров определяется на странице свойств <i>Настройки</i> . Число графиков не превышает трех.

Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Лог.	$Включить\ ФБ = 1.$	
Информационные выходы	Калорифер не используется	Лог.	$Включить\ ФБ = 1$ и $ФБ\ используется = 0.$	<p>Водной нагреватель.</p> <p>Состояние: Остановлен</p>
	Калорифер запущен	Лог.	$Включить\ ФБ = 1$ (во всех последующих выходах в логике вход $Включить\ ФБ = 1$. В противном случае, ФБ на мнемосхеме отображаться не будет), $ФБ\ используется = 1$, $Переходный\ режим = 0.$	<p>Водной калорифер в данное время года не используется.</p> <p>Водной нагреватель.</p> <p>Состояние: Запущен</p>
	Калорифер запуск	Лог.	$ФБ\ используется = 1$, $Переходный\ режим = 1.$	<p>Нагрев: 0%</p> <p>Состояние: Прогрев</p>
	Калорифер перегрев	Лог.	$ФБ\ используется = 1$, $Перегрев = 1.$	<p>Прогрев калорифера до заданной температуры.</p> <p>Мигает надпись Прогрев</p> <p>Состояние: Запущен</p> <p>Ограничение температуры обратной воды не выше установленного предела.</p>

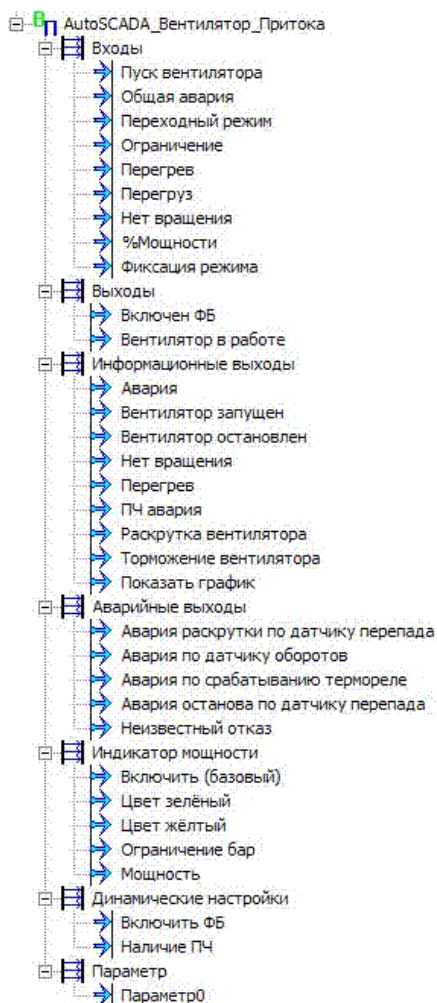
Имя	Тип	Зависимости (Формулы, связи)	Примечание
Калорифер недогрев	Лог.	$ФБ\ используется = 1, Недогрев = 1.$	 <p>Состояние: Запущен</p>
Калорифер проворот	Лог.	$ФБ\ используется = 1, Включить\ калорифер = 1, Проворот = 1.$	<p>Ограничение температуры обратной воды не ниже установленного предела.</p>  <p>Состояние: Запущен</p>
Нагрев секция 1	Лог.	Принимает значение 1 при значении входа %Прогрева < 33%.	<p>Активен режим мягкого запуска.</p>  <p>Состояние: Запущен</p>
Нагрев секция 2	Лог.	Принимает значение 1 при значении входа %Прогрева от 33% до 66%.	 <p>Состояние: Запущен</p>
Нагрев секция 3	Лог.	Принимает значение 1 при значении входа %Прогрева > 66%.	 <p>Состояние: Запущен</p>
Нет воды	Лог.	$Включить\ калорифер = 1, Нет\ воды = 1.$	 <p>Состояние: Остановлен</p>
Аварийные выходы	Общая авария	Лог.	<p>$Авария = 1.$</p> <p>Управление калорифером отключено по причине отсутствия необходимого давления воды в трубопроводах.</p> <p>Авария</p>  <p>Состояние: Авария</p>

Имя	Тип	Зависимости (Формулы, связи)	Примечание	
Аварии	Лог.	$Общая\ авария = 1, Заморозка = 1, Ошибка\ выполнения = 1.$	 <p>Состояние: Авария</p> <p>Множественные сигналы об опасности обмерзания калорифера.</p>	
Авария Т стат.	Лог.	$Общая\ авария = 1, Ошибка\ выполнения = 1, Ошибка\ внешняя = 1.$	 <p>Состояние: Авария</p> <p>Активен режим защиты от обмерзания водяного нагревателя. Падение температуры воздуха за калорифером (термостат).</p>	
Авария Т обр.	Лог.	$Общая\ авария = 1, Заморозка = 1.$	 <p>Состояние: Авария</p> <p>Активен режим защиты от обмерзания водяного нагревателя. Аварийно низкая температура воды на выходе из калорифера.</p>	
Авария Т прит.	Лог.	$Общая\ авария = 1, Ошибка\ внешняя = 1.$	 <p>Состояние: Авария</p> <p>Активен режим защиты от обмерзания водяного нагревателя. Падение температуры воздуха за калорифером (датчик температуры).</p>	
Выходы насоса	Насос работает	Лог.	$Насос\ Работа = 1.$	 <p>Состояние: Запущен</p>
	Авария насоса	Лог.	$Насос\ Авария = 1.$	 <p>Состояние: Авария</p> <p>Перегрузка насоса калорифера. Насос отключен, управление калорифером отключено.</p>
Индикатор мощности	Включить (базовый)	Лог.	$Общая\ авария = 1\ или\ ФБ\ используется = 1\ и\ Нет\ воды = 0.$	
	Цвет зеленый	Лог.	$Включить\ (базовый) = 1, Ограничение = 0.$	
	Цвет желтый	Лог.	$Включить\ (базовый) = 1, Ограничение = 1.$	
	Нагрев	Лог.	$\%Прогрева$	Отображается текущее значение мощности нагрева

Наличие групп информационных, насосов, аварийных выходов и выходов индикатора мощности, настраивается на странице свойств ФБ Настройки.

Вентилятор притока

Вид функционального блока *Вентилятор притока* в дереве объекта:



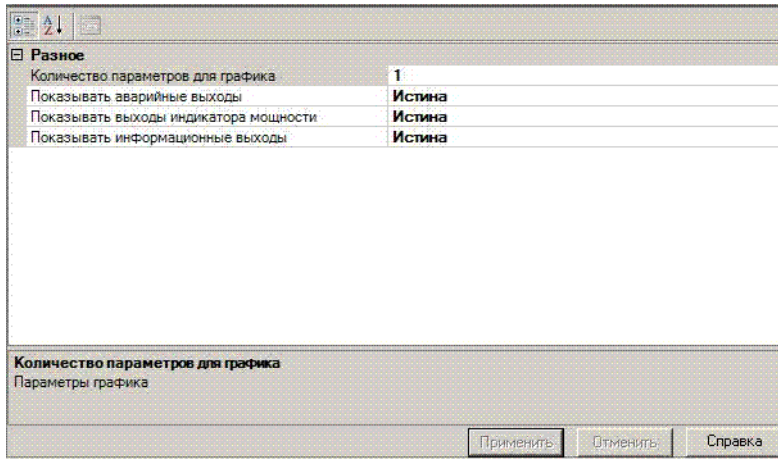
Визуальный функциональный блок *Вентилятор притока* предназначен для определения состояния увлажнителя данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы вентилятора и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображения (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы вентилятора, аварий и параметра мощности в виде прогресс бара.
- Отдельное рор-уп окно с изображением вентилятора, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы вентилятора, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние вентилятора в каждый момент времени.

Страница свойств *Настройки* ФБ вентилятор притока выглядит следующим образом:

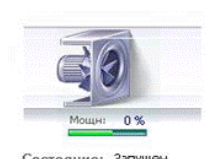
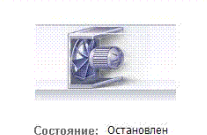
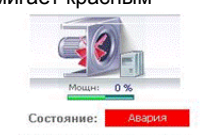
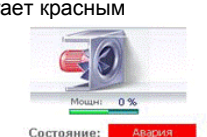
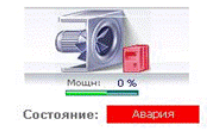
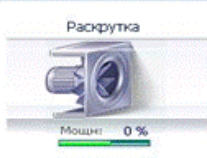
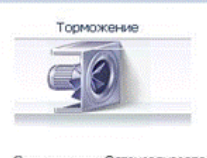







Элементы управления:

- *Количество параметров для графика* - параметры графика.
- Если в поле *Показывать аварийные выходы* установлено значение *Истина*, то в дереве объектов появляется группа аварийных выходов.
- Если в поле *Показывать выходы индикатора мощности* установлено значение *Истина*, то в дереве объектов появляется группа *Индикаторы мощности*.
- Если в поле *Показывать информационные выходы* установлено значение *Истина*, то в дереве объектов появляется группа *Информационные выходы*.

Входы и выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Пуск вентилятора	Логический	от OPC	Команда. При значении 1 запускает вентилятор.
	Общая авария	Логический	от OPC	Команда. Если 1, то на схеме появляется аварийное предупреждение этого ФБ.
	Переходный режим	Логический	от OPC	Команда предназначена для определения режимов работы вентилятора: запуска или останова.
	Ограничение	Логический	от OPC	Команда, используемая для индикации режима ограничения мощности вентилятора.
	Перегрев	Логический	от OPC	Срабатывание термореле защиты двигателя.
	Перегруз	Логический	от OPC	Аварийный сигнал от частотного преобразователя
	Нет вращения	Логический	от OPC	Отсутствует подтверждение вращения вентилятора
Выходы	%Мощности	Целый	от OPC	Отображает мощность работы вентилятора.
	Фиксация режима	Логический	от OPC	Используется только в расчетах.
	Включен ФБ	Логический	1 - Включить ФБ.	Информирует о состоянии ФБ.
Информационные выходы ФБ	Вентилятор в работе	Логический	1 - Включить ФБ и (не(Вентилятор остановлен или Раскрутка вентилятора или Торможение вентилятора).	Информирует о состоянии ФБ.
	Авария	Логический	1 - Общая авария.	Сообщение Авария. 
	Вентилятор запущен	Логический	1 - Пуск вентилятора и (не Общая авария).	Вращение вентилятора, сообщение Запущен.  Состояние: Авария

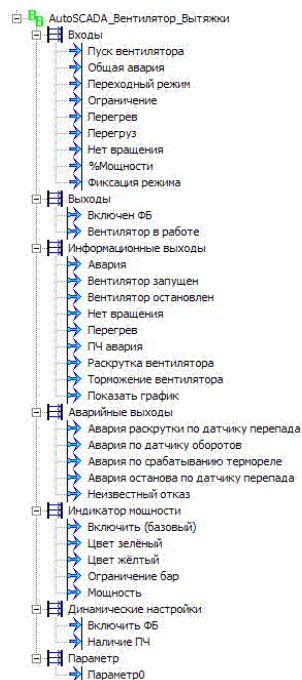
Имя	Тип	Зависимости (Формулы, связи)	Примечание
			Вентилятор вытяжки  Состояние: Запущен
Вентилятор остановлен	Логический	1 - не (Пуск вентилятора или Переходный режим) или Общая авария.	Вращение вентилятора, сообщение Остановлен . Вентилятор притока
			 Состояние: Остановлен
Нет вращения	Логический	1 - Нет вращения и Общая авария.	Изображение крыльчатки мигает красным
			 Состояние: Авария
Перегрев	Логический	1 - Перегрев и Общая авария.	Изображение двигателя мигает красным
			 Состояние: Авария
			Останов по причине срабатывания термореле защиты двигателя.
ПЧ авария	Логический	1 - Наличие ПЧ и Перегруз.	Изображение ПЧ мигает красным
			 Состояние: Авария
			Принят аварийный сигнал от регулятора оборотов вентилятора (частотного преобразователя)
Раскрутка вентилятора	Логический	1 - Пуск вентилятора и Переходный режим и (не Общая авария).	Сообщение Запускается , сообщение Раскрутка - мигает.
			Вентилятор вытяжки Раскрутка  Состояние: Запускается
Торможение вентилятора	Логический	1 - (не Пуск вентилятора) и Переходный режим и (не Общая авария).	Сообщение Торможение , сообщение Останавливается - мигает.
			Вентилятор вытяжки Торможение  Состояние: Останавливается

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Аварийные выходы ФБ	Авария раскрутки по датчику перепада	Логический	1 - Нет вращения и Общая авария и Фиксация режима и (не Перегрев) и (не Перегруз).	 <p>Состояние: Авария</p> <p>Нет подтверждения раскрутки вентилятора от датчика перепада давления (PDS). Это может быть вызвано неработоспособностью датчика или обрывом ремня привода крыльчатки</p>
	Авария по датчику оборотов	Логический	1 - Наличие ПЧ и Общая авария и Перегруз и (не Перегрев) и (не Нет вращения).	 <p>Состояние: Авария</p> <p>Принят аварийный сигнал от регулятора оборотов вентилятора (частотного преобразователя)</p>
	Авария по срабатыванию термореле	Логический	1 - Перегрев и Общая авария и (не Перегруз) и (не Нет вращения).	 <p>Состояние: Авария</p> <p>Останов по причине срабатывания термореле защиты двигателя.</p>
	Авария останова по датчику перепада	Логический	1 - Нет вращения и Общая авария и (не Фиксация режима) и (не Перегрев) и (не Перегруз).	 <p>Состояние: Авария</p> <p>Нет подтверждения остановки вентилятора от датчика перепада давления (PDS). Это может быть вызвано неработоспособностью датчика или залипанием пускателя, включающего вентилятор</p>
	Неизвестный отказ	Логический	1 - Общая авария и (не Авария раскрутки по датчику перепада или Авария по датчику оборотов или Авария по срабатыванию термореле или Авария останова по датчику перепада)).	 <p>Состояние: Авария</p> <p>Зафиксировано наличие неизвестного отказа.</p>
Индикатор мощности	Включить (базовый)	Логический	1 - Пуск вентилятора.	На мнемосхеме отображается индикатор <i>Мощность</i> .
	Цвет зеленый	Логический	1 - Включить (базовый) и (не Ограничение).	На мнемосхеме отображается индикатор <i>Мощность</i> .
	Цвет желтый	Логический	1 - Включить (базовый) и Ограничение.	На мнемосхеме отображается индикатор <i>Мощность</i> .
Мощность	Целый	%Мощности.	На мнемосхеме отображается индикатор <i>Мощность</i> - процент заливки. Индикация цифрами.	
Динамические настройки	Включить ФБ	Логический	от OPC.	Начальное состояние - 0, при этом ФБ в режиме RT должен полностью отсутствовать в системе (визуально, сообщение и пр.)
	Наличие ПЧ	Логический	от OPC.	Начальное состояние - 0, 1 - добавить изображение ПЧ

Наличие групп выходов настраивается на странице свойств ФБ Настройки.

Вентилятор вытяжки

Вид функционального блока *Вентилятор вытяжки* в дереве объекта:



Визуальный функциональный блок *Вентилятор вытяжки* предназначен для определения состояния увлажнителя данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы вентилятора и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображения (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы вентилятора, аварий и параметра мощности в виде прогресс бара;
- Отдельное pop-up окно с изображением вентилятора, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы вентилятора, аварий, параметра мощности в виде прогресс бара, аварийные сообщения и график на 1-3 параметра.

Логика работы функционального блока следующая – в зависимости от комбинации входных сигналов определяется и отображается состояние вентилятора в каждый момент времени.

Страница свойств *Настройки* ФБ вентилятор вытяжки выглядит следующим образом:

Разное	
Количество параметров для графика	1
Показывать аварийные выходы	Истина
Показывать выходы индикатора мощности	Истина
Показывать информационные выходы	Истина

Количество параметров для графика
Параметры графика


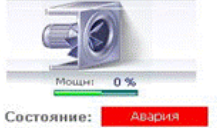
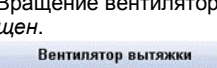

Приложить Отменить Справка

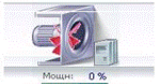







Элементы управления:



- *Количество параметров для графика* - параметры графика;

- Если в поле *Показывать аварийные выходы* установлено значение *Истина*, то в дереве объектов появляется группа *Аварийных выходов*;
- Если в поле *Показывать выходы индикатора мощности* установлено значение *Истина*, то в дереве объектов появляется группа *Индикаторы мощности*;
- Если в поле *Показывать информационные выходы* установлено значение *Истина*, то в дереве объектов появляется группа *Информационные выходы*.

Входы и выходы Вентилятора:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Пуск вентилятора	Логический	от OPC	Команда. При значении <i>Истина</i> запускает вентилятор.
	Общая авария	Логический	от OPC	Команда. Если <i>Истина</i> , то на мнемосхеме появляется аварийное предупреждение этого ФБ.
	Переходный режим	Логический	от OPC	Команда предназначена для определения режимов работы вентилятора: запуска или останова.
	Ограничение	Логический	от OPC	Команда, используемая для индикации режима ограничения мощности вентилятора.
	Перегрев	Логический	от OPC	Срабатывание термореле защиты двигателя.
	Перегруз	Логический	от OPC	Аварийный сигнал от частотного преобразователя
	Нет вращения	Логический	от OPC	Отсутствует подтверждение вращения вентилятора
	%Мощности	Целый	от OPC	Отображает мощность работы вентилятора.
Выходы	Фиксация режима	Логический	от OPC	Используется только в расчетах.
	Включен ФБ	Логический	1 - Включить ФБ.	Информирует о состоянии ФБ.
Информационные выходы ФБ	Вентилятор в работе	Логический	1 - Включить ФБ и (не(Вентилятор остановлен или Раскрутка вентилятора или Торможение вентилятора).	Информирует о состоянии ФБ.
	Авария	Логический	1 - Общая авария.	Сообщение <i>Авария</i> . 
	Вентилятор запущен	Логический	1 - Пуск вентилятора и (не Общая авария).	Вращение вентилятора, сообщение <i>Запущен</i> . 
	Вентилятор остановлен	Логический	1 - не (Пуск вентилятора или Переходный режим) или Общая авария.	Вращение вентилятора, сообщение <i>Остановлен</i> . 
	Нет вращения	Логический	1 - Нет вращения и Общая	Изображение крыльчатки мигает красным 

Имя	Тип	Зависимости (Формулы, связи)	Примечание
		авария.	
Перегрев	Логический	1 - <i>Перегрев</i> и <i>Общая авария</i> .	 <p>Состояние: Авария</p> <p>Нет подтверждения остановки вентилятора от датчика перепада давления (PDS). Это может быть вызвано неработоспособностью датчика или залипанием пускателя, включающего вентилятор</p> <p>Изображение двигателя мигает красным.</p>  <p>Состояние: Авария</p>
ПЧ авария	Логический	"1" - <i>Наличие ПЧ</i> и <i>Перегруз</i> .	<p>Изображение ПЧ мигает красным.</p> <p>Останов по причине срабатывания термореле защиты двигателя.</p>  <p>Состояние: Авария</p> <p>Принят аварийный сигнал от регулятора оборотов вентилятора (частотного преобразователя)</p>
Раскрутка вентилятора	Логический	1 - <i>Пуск вентилятора</i> и <i>Переходный режим</i> и (не <i>Общая авария</i>).	<p>Сообщение <i>Запускается</i>, сообщение <i>Раскрутка</i> - мигает.</p> <p>Вентилятор вытяжки</p>  <p>Состояние: Запускается</p>
Торможение вентилятора	Логический	1 - (не <i>Пуск вентилятора</i>) и <i>Переходный режим</i> и (не <i>Общая авария</i>).	<p>Сообщение <i>Торможение</i>, сообщение <i>Останавливается</i> - мигает.</p> <p>Вентилятор вытяжки</p>  <p>Состояние: Останавливается</p>
Аварийные выходы ФБ	Авария раскрутки по датчику перепада	1 - <i>Нет вращения</i> и <i>Общая авария</i> и <i>Фиксация режима</i> и (не <i>Перегрев</i>) и (не <i>Перегруз</i>).	 <p>Состояние: Авария</p> <p>Нет подтверждения раскрутки вентилятора от датчика перепада давления (PDS). Это может быть вызвано неработоспособностью датчика или обрывом ремня привода крыльчатки</p>
	Авария по датчику оборотов	1 - <i>Наличие ПЧ</i> и <i>Общая авария</i> и <i>Перегруз</i> и (не <i>Перегрев</i>) и (не <i>Нет вращения</i>).	 <p>Состояние: Авария</p> <p>Принят аварийный сигнал от регулятора оборотов вентилятора (частотного преобразователя)</p>
	Авария по срабатыванию термореле	1 - <i>Перегрев</i> и <i>Общая авария</i> и (не <i>Перегруз</i>) и (не <i>Нет вращения</i>).	 <p>Состояние: Авария</p> <p>Останов по причине срабатывания термореле защиты двигателя.</p>

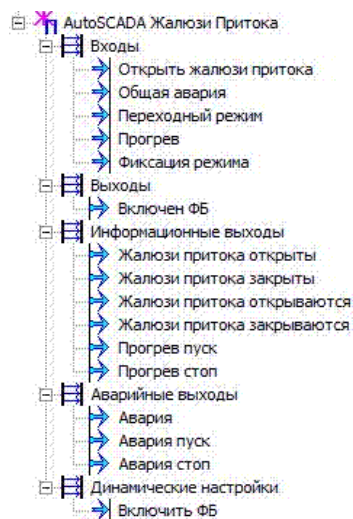
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Индикатор мощности	Авария остановки по датчику перепада	Логический	1 - Нет вращения и Общая авария и (не Фиксация режима) и (не Перегрев) и (не Перегруз).	 <p>Состояние: Авария</p> <p>Нет подтверждения остановки вентилятора от датчика перепада давления (PDS). Это может быть вызвано неработоспособностью датчика или залипанием пускателя, включающего вентилятор.</p>
	Неизвестный отказ	Логический	1 - Общая авария и (не Авария раскрутки по датчику перепада или Авария по датчику оборотов или Авария по срабатыванию термореле или Авария остановки по датчику перепада)).	 <p>Состояние: Авария</p> <p>Зафиксировано наличие неизвестного отказа.</p>
	Включить (базовый)	Логический	1 - Пуск вентилятора.	На мнемосхеме отображается индикатор Мощность .
	Цвет зеленый	Логический	1 - Включить (базовый) и (не Ограничение).	На мнемосхеме отображается индикатор Мощность .
	Цвет желтый	Логический	1 - Включить (базовый) и Ограничение.	На мнемосхеме отображается индикатор Мощность .
Динамические настройки	Мощность	Целый	%Мощности.	На мнемосхеме отображается индикатор Мощность - процент заливки. Индикация цифрами.
	Включить ФБ	Логический	от OPC	Начальное состояние - 0, при этом ФБ в режиме RT должен полностью отсутствовать в системе (визуально, сообщение и пр.).
	Наличие ПЧ	Логический	от OPC	Начальное состояние - 0, 1 - добавить изображение ПЧ.

Наличие групп выходов настраивается на странице свойств ФБ Настройки.

- ❖ **Важно!** Логика работы ФБ Вентилятор притока и ФБ Вентилятор вытяжки идентична. Различие состоит лишь в визуальном отображении этих элементов.

Жалюзи притока

Вид функционального блока Жалюзи притока в дереве Объекта:



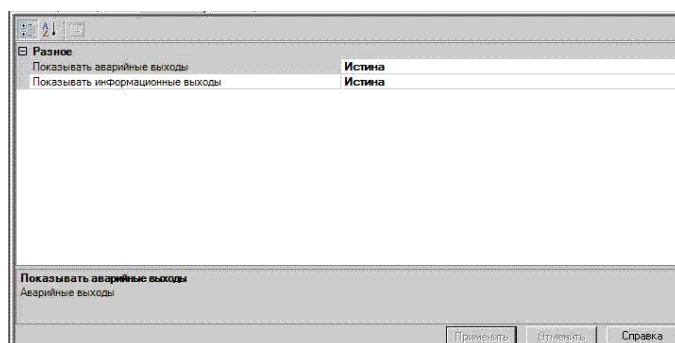
Визуальный функциональный блок *Жалюзи притока* предназначен для определения состояния заслонок данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы жалюзи и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создаётся перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы жалюзи и аварий;
- Отдельное pop-up окно с изображением жалюзи, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы увлажнителя, аварий и аварийные сообщения.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние жалюзи в каждый момент времени.

Страница свойств *Настройки* ФБ жалюзи выглядит следующим образом:


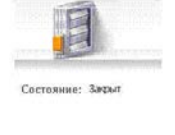
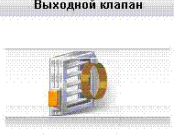


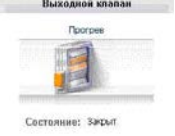




Элементы управления:

- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний жалюзи.

Входы и выходы Жалюзи притока:

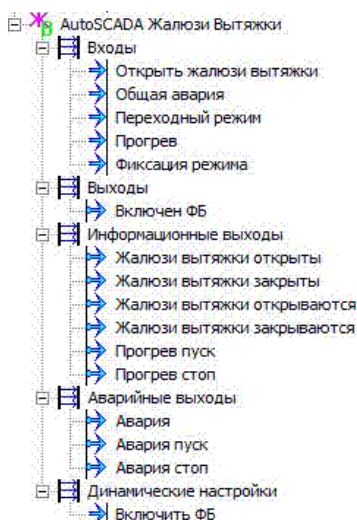
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Открыть жалюзи притока (вытяжки)	Логический	от OPC	Сигнал на открытие жалюзи.
	Общая авария	Логический	от OPC	Аварийный сигнал жалюзи
	Переходный режим	Логический	от OPC	Сигнал определения переходного режима жалюзи: открываются или закрываются.
	Прогрев	Логический	от OPC	Сигнал определяет режим прогрева сегментов заслонки.
	Фиксация режима	Логический	от OPC	Фиксирует режим работы жалюзи: открыты/закрыты, открываются/закрываются.
Динамические настройки ФБ	Включить ФБ	Логический	от OPC	Информирует о текущем состоянии объекта. Значение 1 определяет, что этот элемент используется в конкретной вентиляционной установке. Состояние 0 (выкл.) данного входа скрывает (делает невидимым) всякое представление жалюзи. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно.

	Имя	Тип	Зависимости (Формулы, связи)	Примечание	
Выходы	Включен ФБ	Логический	Принимает значение 1 при значении 1 входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.	
Информационные выходы	Жалюзи притока открыты	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1</i> , <i>Открыть жалюзи притока = 1</i> и <i>Фиксация режима = 1</i> .	 <p>Состояние: Открыт</p>	
	Жалюзи притока закрыты	Логический	Принимает значение 1 при значении входов <i>Включить ФБ = 1</i> (Во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет), <i>Открыть жалюзи притока = 0</i> и <i>Фиксация режима = 1</i> .	 <p>Состояние: Закрыт</p>	
	Жалюзи притока открываются	Логический	Принимает значение 1 при значении входов <i>Открыть жалюзи притока = 1</i> , <i>Фиксация режима = 1</i> и <i>Переходный режим = 1</i> .	 <p>Состояние: Открывается</p>	
	Жалюзи притока закрываются	Логический	Принимает значение 1 при значении входов <i>Открыть жалюзи притока = 0</i> , <i>Фиксация режима = 1</i> и <i>Переходный режим = 1</i> .	 <p>Состояние: Закрывается</p>	
	Прогрев пуск	Логический	Принимает значение 1 при значении входа <i>Прогрев = 1</i> у открытых жалюзи.	 <p>Состояние: Открыт</p> <p>Прогрев сегментов заслонки для предотвращения заклинивания подвижных частей наводками льдом</p>	
	Прогрев стоп	Логический	Принимает значение 1 при значении входа <i>Прогрев = 1</i> и <i>Фиксация режима = 0</i> и <i>Открыть жалюзи притока = 0</i> .	 <p>Состояние: Закрыт</p> <p>Прогрев сегментов заслонки для предотвращения заклинивания подвижных частей наводками льдом</p>	
Аварийные выходы	Авария	Логический	$Общая\ авария = 1$.	Используется для обработки аварийных сигналов. Возникает при поступлении сигнала об общей аварии при открытии жалюзи.	
	Авария пуск	Логический	$Открыть\ жалюзи\ притока = 1$ и $Общая\ авария = 1$ и $Фиксация\ режима = 1$.		 <p>Состояние: Авария</p> <p>Нет подтверждения, что клапан открылся. Это может быть вызвано неработоспособностью датчиков положения заслонки или серверов.</p>
	Авария стоп	Логический	$Общая\ авария = 1$ и $Фиксация\ режима = 0$.		Возникает при поступлении сигнала об общей аварии при закрытии жалюзи.  <p>Состояние: Авария</p> <p>Нет подтверждения, что клапан закрылся. Это может быть вызвано неработоспособностью датчиков положения заслонки или серверов.</p>

Наличие групп выходов настраивается на странице свойств ФБ *Настройки*.

Жалюзи вытяжки

Вид функционального блока *Жалюзи вытяжки* в дереве объекта:



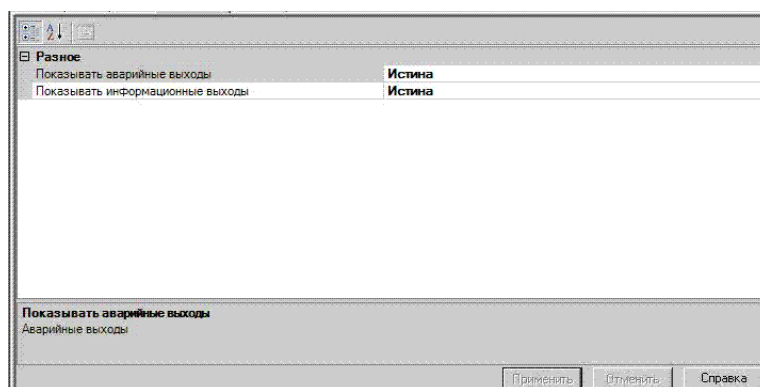
Визуальный функциональный блок *Жалюзи вытяжки* предназначен для определения состояния заслонок данного типа. ФБ также предоставляет возможность отображения на мнемосхеме рассматриваемого элемента вентиляционной установки, привязанного к этому ФБ. ФБ предназначен для визуализации технологических параметров работы жалюзи и не предусматривает функций управления.

ФБ имеет два типа не редактируемых изображений (окон):

- Изображение на общей технологической мнемосхеме в режимах DT и RT – создается перетаскиванием ФБ из дерева объектов на мнемосхему. Имеет ограниченный функциональный набор – только анимацию работы жалюзи и аварий;
- Отдельное рор-ур окно с изображением жалюзи, доступное только в режиме RT по нажатию на первое изображение. Имеет полный функциональный набор – анимация работы увлажнителя, аварий и аварийные сообщения.

Логика работы функционального блока следующая - в зависимости от комбинации входных сигналов определяется и отображается состояние жалюзи в каждый момент времени.

Страница свойств *Настройки* ФБ жалюзи вытяжки выглядит следующим образом:



Элементы управления:




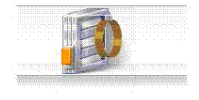

- *Показывать аварийные выходы* – определяет наличие выходов, относящихся к авариям;
- *Показывать информационные выходы* – определяет наличие выходов, относящихся к отображению состояний жалюзи.

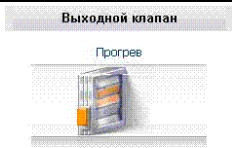
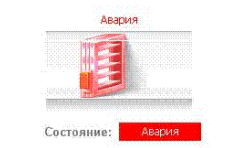
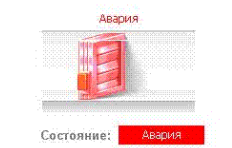
Входы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Входы	Открыть жалюзи вытяжки	Логический	от OPC	Сигнал на открытие жалюзи.
	Общая авария	Логический	от OPC	Аварийный сигнал жалюзи

Динамические настройки ФБ	Переходный режим	Логический	от OPC	Сигнал определения переходного режима жалюзи: открываются или закрываются. Сигнал определяет режим прогрева сегментов заслонки. Фиксирует режим работы жалюзи: открыты/закрыты, открываются/закрываются. Информирует о текущем состоянии объекта. Значение <i>1</i> определяет, что этот элемент используется в конкретной вентиляционной установке. Состояние <i>0</i> (<i>выкл.</i>) данного входа скрывает (делает невидимым) всякое представление жалюзи. Изменение состояния входа <i>Включить ФБ</i> меняет состояние выхода <i>Включен ФБ</i> соответственно.
	Прогрев	Логический	от OPC	
	Фиксация режима	Логический	от OPC	
	Включить ФБ	Логический	от OPC	

Выходы ФБ:

	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Выходы	Включен ФБ	Логический	Принимает значение <i>1</i> при значении <i>1</i> входа <i>Включить ФБ</i> .	Информирует о состоянии ФБ.
Информационные выходы	Жалюзи вытяжки открыты	Логический	Принимает значение <i>1</i> при значении входов <i>Включить ФБ = 1</i> , <i>Открыть жалюзи вытяжки = 1</i> и <i>Фиксация режима = 1</i> .	<p>Выходной клапан</p>  <p>Состояние: Открыт</p> <p>Выходной клапан</p>
	Жалюзи вытяжки закрыты	Логический	Принимает значение <i>1</i> при значении входов <i>Включить ФБ = 1</i> (Во всех последующих выходах в логике вход <i>Включить ФБ = 1</i> . В противном случае, ФБ на мнемосхеме отображаться не будет), <i>Открыть жалюзи вытяжки = 0</i> и <i>Фиксация режима = 1</i> .	<p>Выходной клапан</p>  <p>Состояние: Закрыт</p>
	Жалюзи вытяжки открываются	Логический	Принимает значение <i>1</i> при значении входов <i>Открыть жалюзи вытяжки = 1</i> , <i>Фиксация режима = 1</i> и <i>Переходный режим = 1</i> .	<p>Выходной клапан</p>  <p>Состояние: Открывается</p>
	Жалюзи вытяжки закрываются	Логический	Принимает значение <i>1</i> при значении входов <i>Открыть жалюзи вытяжки = 0</i> , <i>Фиксация режима = 1</i> и <i>Переходный режим = 1</i> .	<p>Выходной клапан</p>  <p>Состояние: Закрывается</p>
	Прогрев пуск	Логический	Принимает значение <i>1</i> при значении входа <i>Прогрев = 1</i> у открытых жалюзи.	<p>Выходной клапан</p> <p>Прогрев</p>  <p>Состояние: Открыт</p> <p>Прогрев сегментов заслонки для предотвращения заклинивания подвижных частей намерзшим льдом</p>

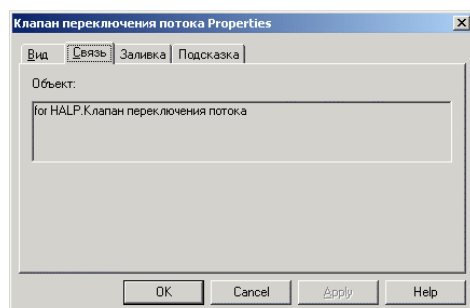
	Имя	Тип	Зависимости (Формулы, связи)	Примечание
Аварийные выходы	Прогрев стоп	Логический	Принимает значение 1 при значении входа <i>Прогрев</i> = 1 и <i>Фиксация режима</i> = 0 и <i>Открыть жалюзи вытяжки</i> = 0.	 <p>Состояние: Закр.</p> <p>Прогрев сегментов заслонки для предотвращения заклинивания подвижных частей намерзшим льдом</p>
	Авария	Логический	<i>Общая авария</i> = 1.	Используется для обработки аварийных сигналов. Возникает при поступлении сигнала об общей аварии при открытии жалюзи.
	Авария пуск	Логический	<i>Открыть жалюзи вытяжки</i> = 1 и <i>Общая авария</i> = 1 и <i>Фиксация режима</i> = 1.	
	Авария стоп	Логический	<i>Общая авария</i> = 1 и <i>Фиксация режима</i> = 0.	 <p>Состояние: Авария</p> <p>Нет подтверждения, что клапан открылся. Это может быть вызвано неработоспособностью датчиков положения заслонки или сервопривода</p>
				 <p>Состояние: Авария</p> <p>Нет подтверждения, что клапан закрылся. Это может быть вызвано неработоспособностью датчиков положения заслонки или сервопривода</p>

Наличие групп выходов настраивается на странице свойств ФБ Настройки.

Общие свойства и настройки

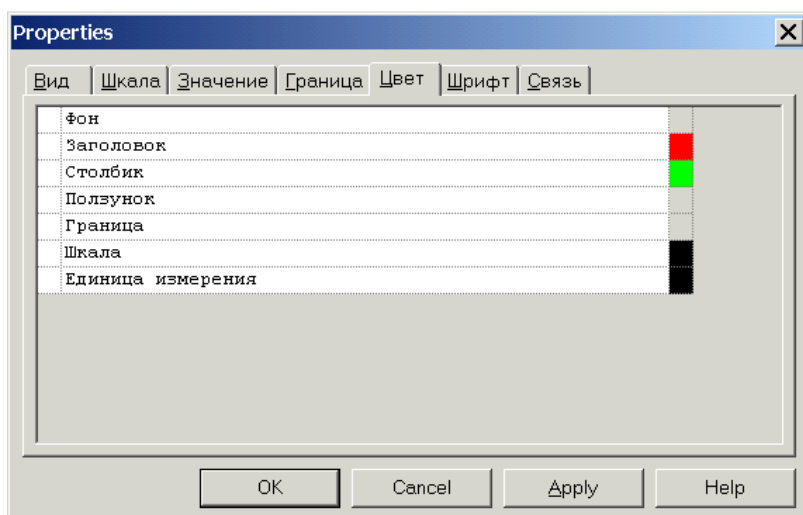
Закладка Связь

На закладке *Связь* отображается связь ВФБ с переменной проекта и путь до этой переменной в дереве:



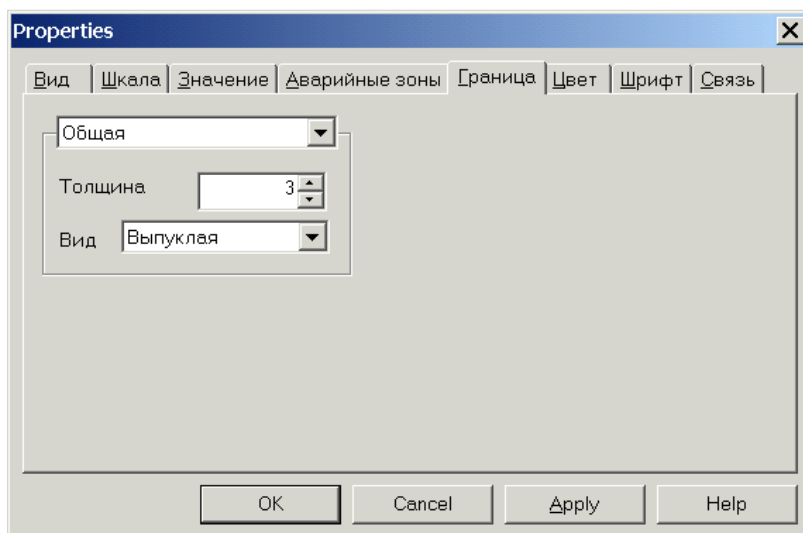
Закладка Цвет

На закладке *Цвет* настраиваются цвета всех элементов ВФБ, отображаемых на мнемосхеме (заголовок, граница, шкала, значение в норме, ползунок и т.д.). Первоначально все элементы имеют свой стандартный цвет. Если необходимо сменить стандартный цвет необходимо щелкнуть мышкой по цветному квадратику напротив названия элемента, чей цвет вы хотите изменить. При этом появится стандартный диалог *Windows* для выбора цвета.



Закладка Граница

На закладке *Граница* настраивается вид границы отображения ВФБ

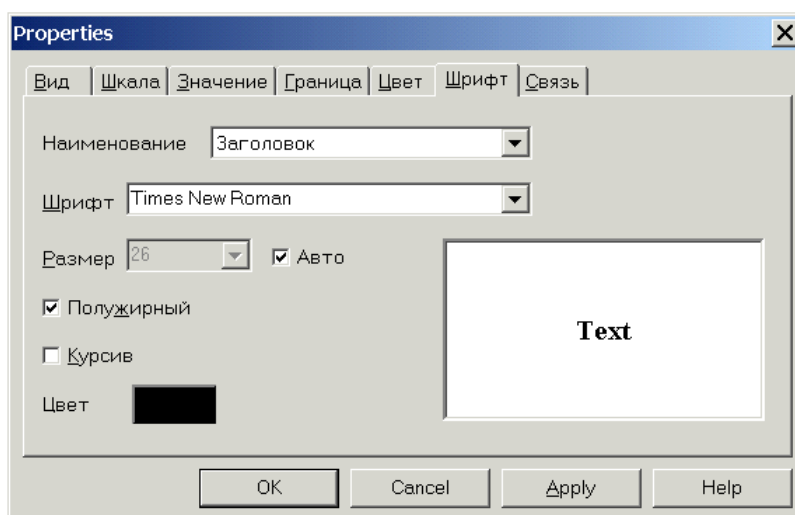


Элементы управления:

Элемент диалога	Описание
Выпадающий список с названиями элементов отображения	Указывает элемент отображения ВФБ, для которой будет настраиваться границы.
Поле ввода <i>Толщина</i>	Задает толщину границы.
Выпадающий список <i>Вид</i>	Задает вид границы выбранного элемента (выпуклая, вогнутая или плоская).

Закладка Шрифт

На закладке *Шрифт* настраиваются атрибуты шрифтов всех элементов ВФБ

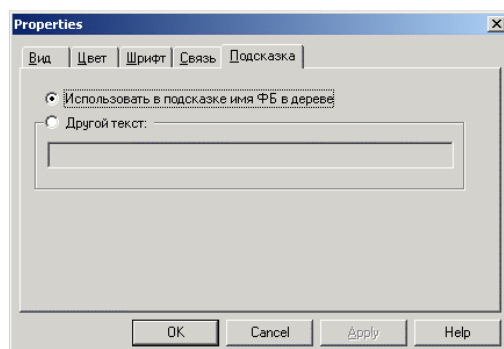


Элементы управления:

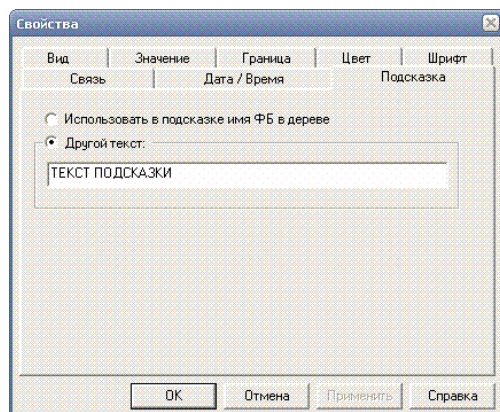
Элемент диалога	Описание
Выпадающий список <i>Наименование</i>	Указывает название элемента, для которой будут производиться настройки шрифта.
Выпадающий список <i>Шрифт</i>	Задаёт тип шрифта.
Поле ввода <i>Размер</i>	Задаёт размер шрифта.
Флажок <i>Авто</i>	Устанавливает автоматическую подгонку размера шрифта.
Флажок <i>Полужирный</i>	Устанавливает атрибут <i>полужирный</i> шрифта.
Флажок <i>Курсив</i>	Устанавливает атрибут <i>курсив</i> .
Кнопка <i>Цвет</i>	Задаёт цвет шрифта (нажатие на кнопку вызывает стандартный диалог Windows для выбора цвета).
Окно просмотра	Показывает пример надписи с указанными атрибутами шрифта.

Закладка Подсказка

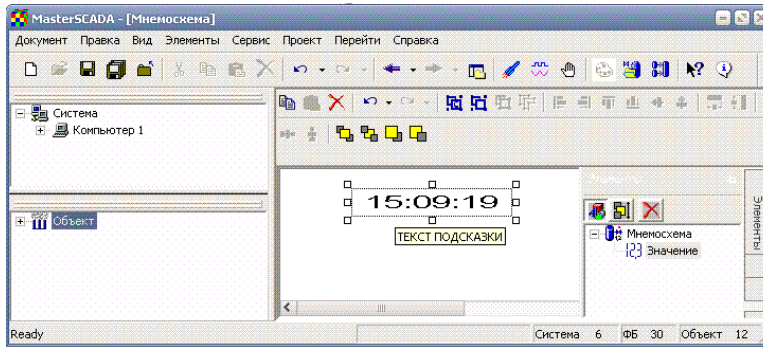
На этой закладке настраивается всплывающая подсказка, которая будет отображаться как в режиме разработки, так и в режиме исполнения при наведении курсора мыши на данный элемент. Подсказка может отображать полный путь к элементу в дереве объектов, или произвольный текст, введенный пользователем.



Например, если ввести в поле *Другой текст* текст всплывающей подсказки:



он отобразится на мнемосхеме:

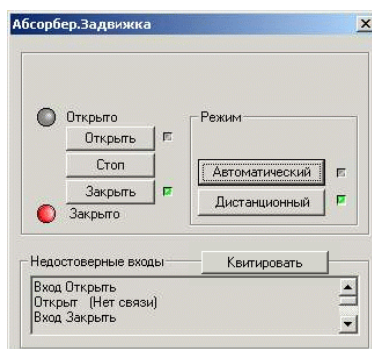


Отображение недостоверных данных

Визуальные функциональные блоки категорий *Исполнительные механизмы*, *Датчики*, *Аппараты*, *ВФБ Таблица* (категории *Управление*), переменные MasterSCADA. Недостоверность данных входов способны отображать только Визуальные функциональные блоки. Данные, пришедшие на входы функционального блока считаются недостоверными, если хотя бы один из входов функционального блока имеет *плохой* OPC-признак качества. Индикация признаков недостоверности зависит от связанного с ФБ элемента мнемосхемы. Для элементов, представляющих данные в виде числовых значений, эта строка подчеркивается, для элементов, имеющих только графическое изображение, оно становится более бледным.

ВФБ категории *Исполнительные механизмы*:

При наличии признака недостоверности данных на входах функционального блока, в нижней части окна оператора появляется список, содержащий информацию об источниках недостоверных данных.



Здесь отображается список входов функционального блока. Напротив каждого входа, данные на котором имеют *плохой* OPC-признак качества, приводится расшифровка этого признака качества. Если хотя бы один вход имеет недостоверные данные, то изображение исполнительного механизма в мнемосхеме становится более бледным. После нажатия кнопки *Квитировать* данные всех входов считаются достоверными (при этом изображение в мнемосхеме перестает быть бледным) до прихода недостоверных данных на какой-либо из входов функционального блока.

ВФБ категории *Датчики*, *Аппараты*, *ВФБ Таблица* и *переменные* MasterSCADA:

ВФБ *График* - если данные на каком-либо из входов функционального блока имеют *плохой* OPC-признак качества, то в таблице параметров *Графика*, соответствующий параметр будет подчеркнут.



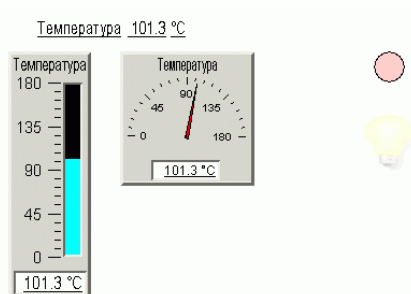
ВФБ *Индикатор состояния* - если данные на входе этого функционального недостоверны, то его изображение в мнемосхеме становится более бледным.

ВФБ *Емкость* и *Емкость с мешалкой* - при настройке изображения ФБ в мнемосхеме на закладке *Цвет* выбирается *Цвет при недостоверности*. Если хотя бы один вход ФБ имеет недостоверные данные, то изображение ФБ в мнемосхеме закрашивается этим цветом.

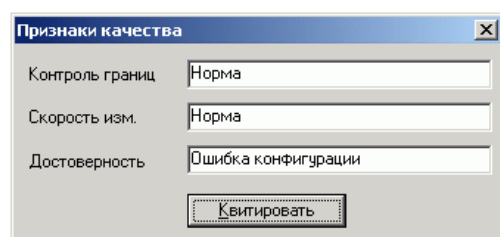
ВФБ *Таблица* - если данные на каком-либо из входов функционального блока имеют *плохой* ОРС-признак качества, то соответствующий параметр будет подчеркнут.

Испаритель			
1	Температура	101.3	°C
2	Давление	261.33	кПа

При использовании *переменных MasterSCADA* в качестве визуальных функциональных блоков, недостоверность отображают следующие элементы: *Редактируемое значение*, *Значение*, *Индикатор значения*, *Стрелочный прибор*, *График*, *Индикатор графический* и *Индикатор мнемонический*. При этом в *Редактируемом значении*, *Значении*, *Индикаторе значения*, *Стрелочном приборе* и *Графике* недостоверные данные подчеркиваются, а у *Индикаторов графического* и *мнемонического* изображение в мнемосхеме становится более бледным.



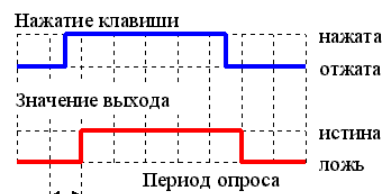
При щелчке левой кнопкой мыши по изображению ВФБ в мнемосхеме, данные на входах которого недостоверны, появляется диалоговое окно, содержащее расшифровку признака недостоверности входа, а также признаки контроля границ и скорости изменения данных входа. Вид этого окна показан ниже:



При нажатии на кнопку *Квитировать*, данные соответствующего входа перестают считаться недостоверными до того момента, пока на вход снова не придут недостоверные данные.

Реакция на нажатие кнопок

Так как опрос функционального блока происходит периодически (что является обязательным условием), то реакция на действия оператора не будет мгновенной. Соответствующий выход будет переведен в *Истина* или *Ложь* при ближайшем опросе функционального блока. Схематично это можно представить в виде временной диаграммы:



Если оператор нажимает клавишу кратковременно, сигнал будет формироваться в течение следующего периода опроса:



Если в течение одного периода оператор успеет нажать клавишу неоднократно, обрабатывается только одно нажатие.

Элементы мнемосхемы

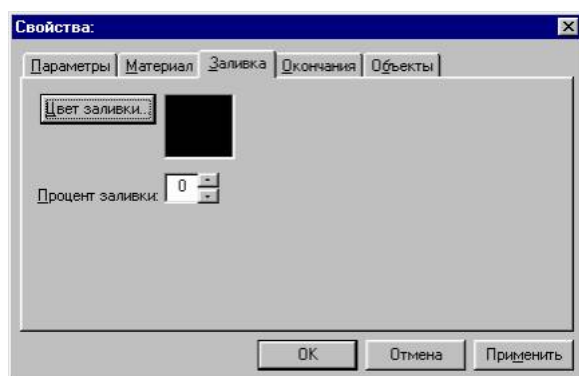
Трехмерные объекты имеют как общие, так и специфические для конкретного типа объекта свойства. К общим свойствам 3D объектов можно получить доступ, открыв диалог *Свойства* и выбрав одну из закладок: *Материал*, *Заливка*, *Окончания* и *Объекты*. Аналогичные группы свойств есть и на палитре настройки.

Помимо общих свойств, у каждого типа объекта есть дополнительные, присущие только ему настройки (специфические свойства). Настройка этих свойств производится на закладке *Параметры* диалога *Свойства*.

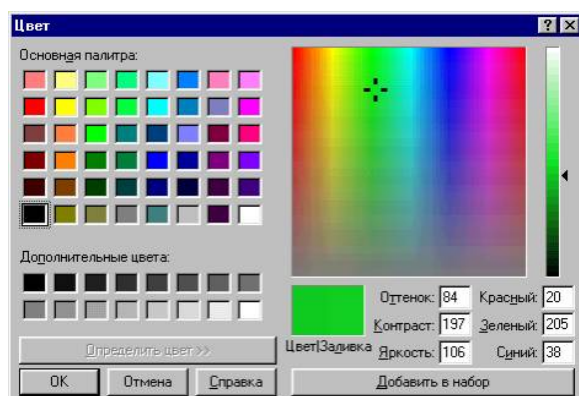
Свойства объектов

Заливка

Свойство **Заливка** служит для изменения параметров заливки материала (уровня жидкости в условном сосуде, образуемом внешним контуром объекта). Заливка определяется двумя настройками: цветом и процентом заливки (отношение высоты цвета заливки к общей высоте объекта).



Изменение цвета заливки производится в стандартном для Windows диалоговом окне *Цвет*, открываемое двойным щелчком на области цвета или нажатием на кнопку *Цвет заливки*. В этом диалоге можно выбрать цвет из основной палитры или создать новый цвет.



Создать новый цвет можно щелкнув в большой правый квадрат. В месте щелчка появится перекрестие и область *Цвет|Заливка* окрасится в выбранный цвет. Оттенок цвета можно изменить при помощи ползунка, расположенного справа от большого квадрата. Определить цвет можно также вводом в поля *Оттенок*, *Контраст*, *Яркость* чисел от 0 до 240 и в поля *Красный*, *Зеленый*, *Синий* чисел от 0 до 255. Кнопка *Добавить* в набор добавляет созданный цвет в дополнительные цвета основной палитры.

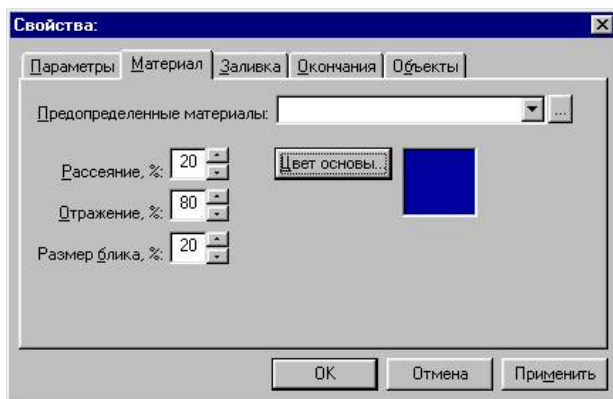
Процент заливки изначально равен 0, т.е. при первоначальной отрисовке заливки нет. Как правило, степень заливки задается динамически в режиме исполнения в соответствии со значением параметра. Для создания заливки введите значение от 1 до 100 в поле *Процент заливки* и нажмите кнопку *ОК*.

Свойство Материал

На закладке **Материал** производится настройка свойств цветового покрытия 3D элементов. Материал характеризуется следующими четырьмя параметрами:

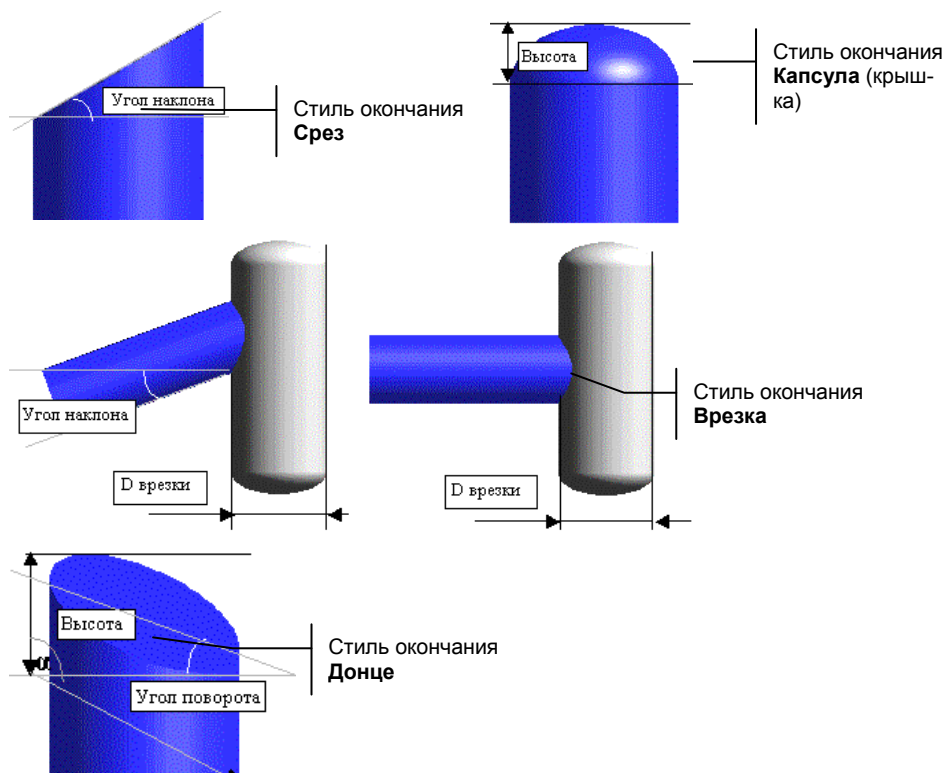
- *Цветом поверхности;*
- *Степенью рассеяния* (коэффициент отражения рассеянного белого света) - чем больше рассеяние, тем ярче становится элемент;
- *Степенью отражения* (коэффициент отражения направленного света, направление и цвет источника задаются в свойствах мнемосхемы) - при полном отсутствии отражения элемент имеет ровную закраску;
- *Размером блика* (размер светового пятна на поверхности элемента).

Эти значения задаются в процентах. Цвет и расположение блика зависят от общих настроек мнемосхемы (цвета и направления луча освещения).



Свойство Окончания

Закладка **Окончания** позволяет изменять начальные и конечные окончания объекта. Для 3D объектов можно использовать 4 стиля окончаний (стили присущи не всем объектам: например, у сегмента поворота нет донца, а у конуса – среза), показанные ниже на примере цилиндра.

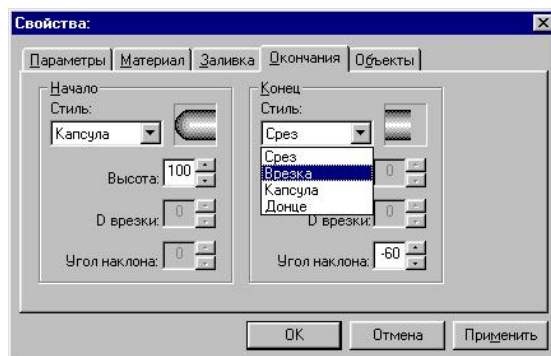


Параметры стилей окончаний могут принимать значения из определенного диапазона:

Параметры окончаний	Диапазон значений
Высота	Для капсулы: от 1 (эквивалентно срезу) до 100 (полусфера) Для донца: от -100 (наклон на нас) до 100 (наклон в обратную сторону).
D (диаметр) врезки	Измеряется в пикселях: от 1 до 999.
Угол наклона	От -60 до 60

Хотя наиболее удобным и быстрым способом задания этих параметров является визуальное проектирование, для более точной настройки используйте диалог *Свойства*.

В диалоге *Свойства* выберите закладку *Окончания*. На ней расположены две группы полей настройки окончаний объекта *Начало* и *Конец*. Стиль окончания выбирается в выпадающем списке *Стиль*, а настройки задаются в полях *Высота*, *D врезки* и *Угол наклона*.

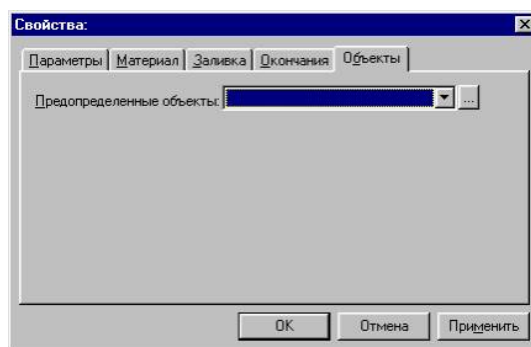


В зависимости от стиля окончания и типа объекта некоторые из этих полей будут недоступны.

Свойство Объекты

Все настройки объекта (его материал, длину, высоту, заливку, окончания и др.) можно сохранять в файле коллекции объектов, а в будущем использовать для создания нового объекта с теми же параметрами. Для совершения операций сохранения и чтения файлов коллекций объектов используется закладка **Объекты**.

Сохранение и чтение коллекций объектов (предопределенных объектов) производится аналогично предопределенным материалам, описание которых указано в следующем разделе.



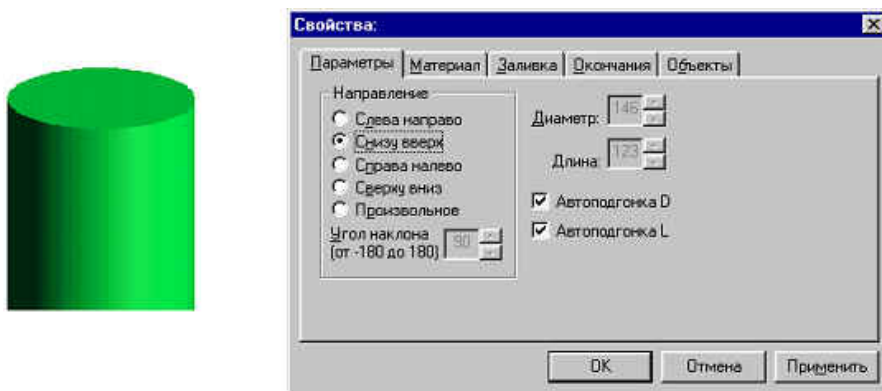
Объёмные объекты

Цилиндр

Объект **Цилиндр** используется для создания несложных объектов, таких как цистерны, бочки, танки, ректификационные колонны, отрезки трубопроводов и т.д.



На рисунке ниже показан вид **Цилиндра** при значениях настроек, заданных на закладке **Параметры**:



Поля настройки закладки **Параметры** цилиндра:

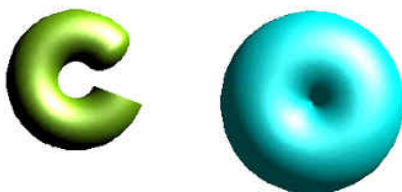
Свойство	Описание
Направление	Направление оси цилиндра: сверху вниз, снизу вверх, слева направо, справа налево и произвольный угол поворота от -180° до 180° .
Диаметр	Диаметр Цилиндра .
Длина	Длина (высота) Цилиндра (без учета длины окончаний).
Автоподгонка D	Автоматическая подгонка диаметра под ширину рамки.
Автоподгонка L	Автоматическая подгонка длины под высоту рамки.

❖ **Важно!** если направление оси **Цилиндра** выбрано произвольным, то флажок **Автоподгонка** недоступен.

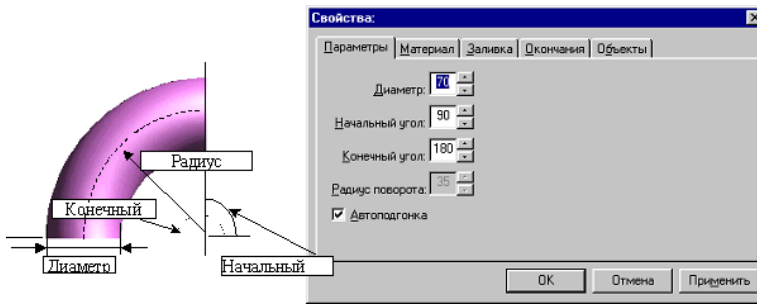
Обратите внимание, что хотя в списке 3D объектов нет сферы, она получается из объекта **Цилиндр** длиной в 1 пиксель и имеющего окончания типа капсула с высотой 100.

Сегмент поворота

Объект **Сегмент поворота** используется для углов поворота трубопроводов, втулок, колец и т.д.



На рисунке ниже показан вид **Сегмента поворота** при значениях настроек, заданных на закладке **Параметры**:



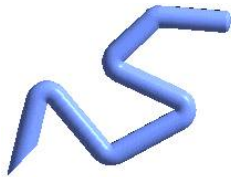
Поля настройки закладки *Параметры* *Сегмента поворота*:

Свойство	Описание
Диаметр	Диаметр <i>Сегмента поворота</i> .
Начальный угол	Угол между горизонтальной плоскостью и начальным окончанием (-180°; 180°).
Конечный угол	Угол между горизонтальной плоскостью и конечным окончанием (-180°; 180°).
Радиус поворота	Радиус дуги <i>Сегмента</i> .
Автоподгонка	Автоматическая подгонка радиуса поворота для точного вписывания элемента в размер рамки.

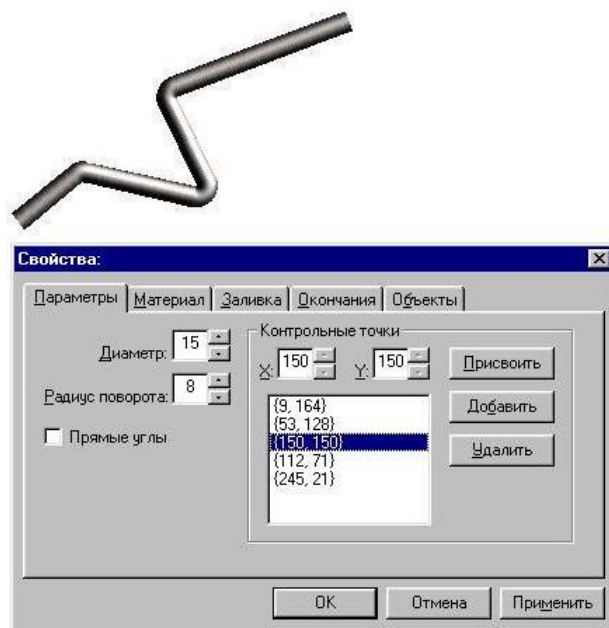
Для данного объекта не определяется угол наклона стилей окончания *Срез* и *Врезка*, отсутствует стиль *Донце*.

Трубопровод

Объект **Трубопровод** позволяет легко создавать трубопроводы различной протяженности, диаметра и конфигурации.



На рисунке ниже показан вид сегмента поворота при значениях настроек, заданных на закладке *Параметры*:



Поля настройки закладки **Параметры** трубопровода:

Свойство	Описание
Диаметр	Диаметр трубы <i>Трубопровода</i> .
Радиус поворота	Аналогично сегменту поворота.
Прямые углы	После установки этого признака все дополнительные отрезки <i>Трубопровода</i> будут прокладываться под прямым углом к предыдущим отрезкам.
Контрольные точки X	Координаты (X; Y) точек построения <i>Трубопровода</i> .
Контрольные точки Y	

Количество контрольных точек *Трубопровода* неограниченно: их можно добавлять, удалять, править. Для этих целей используются кнопки: *Добавить* - добавление новой контрольной точки; *Удалить* - удаление выделенной контрольной точки; *Присвоить* - присвоение значений в полях ввода X и Y координатам выбранной в списке контрольной точки. Но гораздо удобнее добавлять, удалять, править контрольные точки в режиме визуального проектирования, как описано в тексте ниже.

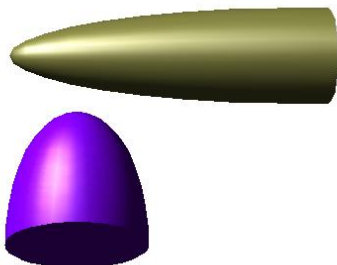
Активные точки бордового цвета показывают контрольные точки. Перемещая эти контрольные точки (нажать левую клавишу мыши и, не отпуская, передвинуть курсор мыши на нужное место) оператор может изменять вид трубы (корректировать углы сгиба).

Чтобы добавить новую контрольную точку выполните двойной щелчок мышью на мнемосхеме (новая точка добавляется в конец или начало трубы, в зависимости от того куда щелкнули мышью). Двойной щелчок на самом *Трубопроводе* добавляет новую контрольную точку в том месте куда щелкнули (между уже существующими точками трубы).

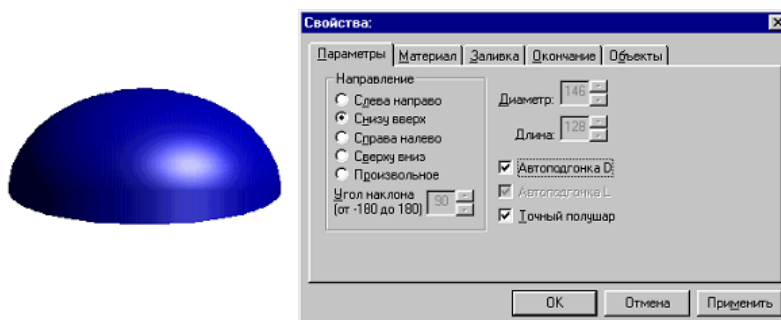
Удаление контрольной точки производится двойным щелчком мышью на уже существующей контрольной точке *Трубопровода*.

Полуэллипсоид

С помощью объекта **Полуэллипсоид** легко создаются различные крышки, элементы химических аппаратов сложной формы и т.п.



На рисунке ниже показан вид *Полуэллипсоида* при значениях настроек, заданных на закладке **Параметры**:



Поля настройки закладки **Параметры** *Полуэллипсоида*:

Свойство	Описание
Направление	Направление оси полуэллипсоида: сверху вниз, снизу вверх, слева направо, справа налево и произвольный угол поворота от -180° до 180° .

Свойство	Описание
Диаметр	Диаметр объекта.
Длина	Длина объекта.
Автоподгонка D	Автоматическая подгонка диаметра полуэллипсоида под размеры рамки. Для произвольного направления объекта свойство недоступно.
Автоподгонка L	Автоматическая подгонка длины полуэллипсоида под размеры рамки. Для произвольного направления объекта свойство недоступно.
Точный полушар	Построение полуэллипсоида с одинаковой длиной и диаметром (полушара).

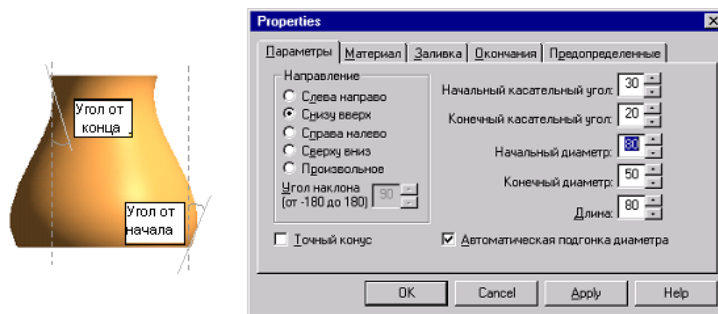
Для данного объекта не определяется угол наклона стилей окончания *Срез*, *Врезка* и *Донце*.

Конус

Объект **Усеченный конус** используется для создания сложных фигур, например, градирни ТЭЦ, домны, емкости и химического аппарата сложной формы (или его части), лампочки и т.п.



На рисунке ниже показан вид *Усеченного конуса* при значениях настроек, заданных на закладке *Параметры*:

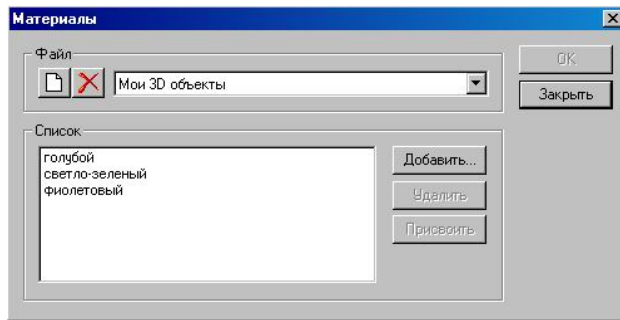


Поля настройки закладки *Параметры* конуса:


Свойство	Описание
Направление	Направление оси конуса: сверху вниз, снизу вверх, слева направо, справа налево и произвольный угол поворота от -180° до 180°.
Точный конус	Поверхность вращения образована прямой линией.
Угол от начала	Угол между направлением оси конуса и касательной к образующей конуса в начальной точке.
Угол от конца	Угол между направлением оси конуса и касательной к образующей конуса в конечной точке.
D начала	Диаметр начальной части конуса.
D конца	Диаметр конечной части конуса.
Длина	Длина конуса (без учета длины окончаний).
Автоподгонка	Автоподгонка диаметра конуса под размер рамки с сохранением формы образующей. Недоступно при <i>произвольном направлении оси</i> .

Предопределенные материалы

Заданные настройки материала можно сохранять в файлах коллекций материалов, а затем использовать, выбирая материал в выпадающем списке по его названию. Для сохранения материала или открытия файла коллекции материалов нажмите кнопку . На экране появится диалог **Материалы**, в котором осуществляется открытие и запись файлов коллекции материалов, а также добавление и изменение материалов в коллекции.

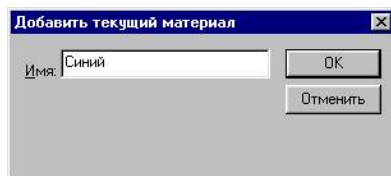


Открытие и создание файлов коллекции

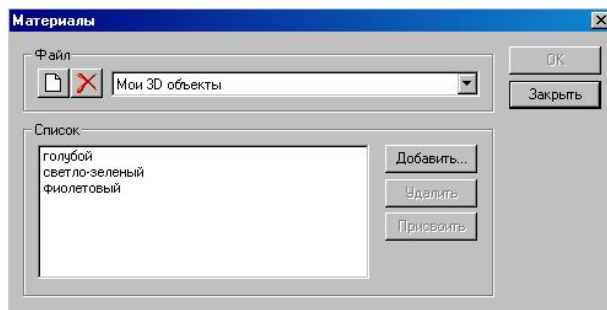
Чтобы открыть существующий файл коллекции материалов, выберите его из выпадающего списка. Если вы хотите создать новую коллекцию, то нажмите кнопку  и в всплывающем окне введите название файла.

Добавление и изменение материалов в коллекции

Для сохранения материала в уже открытой коллекции нажмите кнопку *Добавить*. Появится окно *Добавить текущий материал*.



В этом окне нужно ввести название материала и нажать кнопку *ОК*, после чего этот материал появится в списке материалов диалога *Материалы*.



Если нужно переопределить существующий материал (т.е. изменить его расцветку на расцветку текущего материала), то в диалоге *Материалы* выберите его название в списке материалов и нажмите кнопку *Присвоить*. Теперь этот материал находится в нашей коллекции. Для *Удаления материала* из коллекции выделите материал и нажмите кнопку, после чего материал удаляется из коллекции.

Объекты индикации и управления

График

Визуальный функциональный блок *График* относится к категории *Датчики* и имеет настраиваемое количество входов, соответствующих параметрам графика. Вид функционального блока в дереве объекта представлен на рисунке ниже.

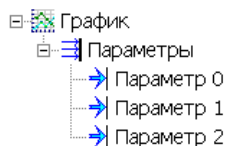


График предназначен для просмотра значений переменных в мнемосхеме. Существует три основных способа создания элемента управления *График* в мнемосхеме:

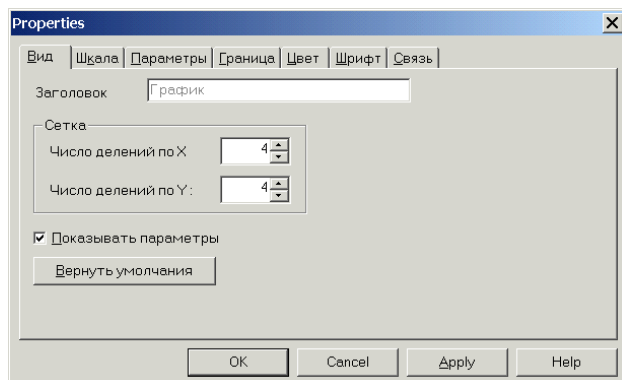
- Вставить ВФБ *График* из палитры функциональных блоков в объект, привязать интересующие переменные к входам *Графика*, и перетащить его в мнемосхему;
- Для просмотра значений аналоговых переменных в виде графика (*Значение*, *Расчет*, *ОРС переменная для чтения*, *Выход ФБ*), можно просто перетащить их в мнемосхему правой кнопкой мыши, и в открывающемся контекстном меню выбрать график;
- Создать элемент управления *График* непосредственно в мнемосхеме. Затем, завести необходимое количество входов мнемосхемы, связанных с переменными MasterSCADA, и динамизировать свойства графика *Значение* от этих входов.

График реализует функции тренда реального времени (зависимость параметра от времени). Может иметь от 1 до 8 параметров. Диапазон шкалы, единица измерения и формат значения каждого параметра наследуются от настроек переменной, привязанной к параметру *Графика* в дереве объекта. Число параметров, отображаемых на *Графике*, задается в проекте на странице Свойства. По умолчанию это всегда один параметр.



ВФБ *График* имеет возможность отображать недостоверность данных входов.

На закладке *Вид* настраивается внешний вид ВФБ *График*



Элемент диалога

Поле ввода *Заголовок*

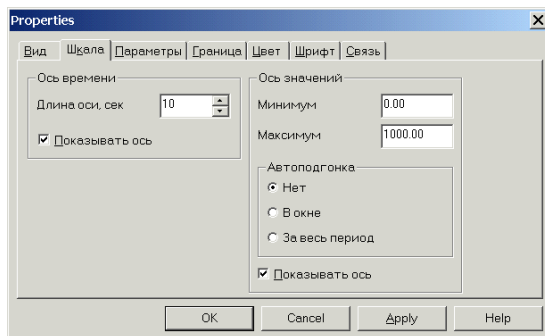
Описание

Текст заголовка наследуется от имени данного экземпляра ВФБ *График* в дереве объекта (название график в дереве объекта может быть изменено), новое имя может быть задано в поле ввода, при отсутствии текста график заголовка не имеет.

Элемент диалога	Описание
Поле ввода <i>Число делений по X</i>	Задаёт число делений графика по оси X.
Поле ввода <i>Число делений по Y</i>	Задаёт число делений графика по оси Y.
Флажок <i>Показывать параметры</i>	Задаёт необходимость отображения таблицы параметров под графиком (таблица содержит для каждого параметра имя, текущее значение и единицу измерения).
Кнопка <i>Вернуть умолчания</i>	Восстанавливает свойства, унаследованные из дерева объекта.

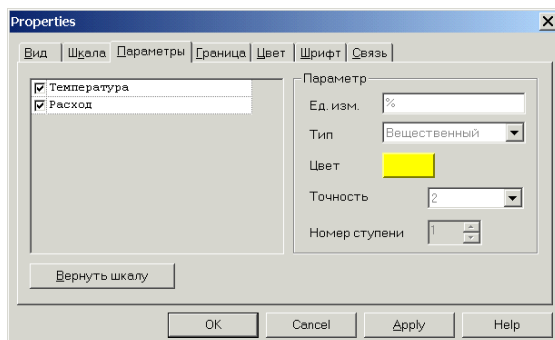
- ❖ **Важно!** При отсутствии связи у выхода ВФБ или работы графического редактора MasterGraph вне среды пакета программ MasterSCADA наследование настроек свойств переменной не работает. В этом случае все свойства графика настраиваются непосредственно на страницах свойств. После установления в дереве связи выхода ВФБ для импорта всех необходимых настроек следует выполнить команду меню *Сервис/Обновить связь*.

На закладке **Шкала** настраиваются оси графика ВФБ.



Элемент диалога	Описание
Ось времени	
Поле ввода - длина оси, сек	Задаёт диапазон оси времени графика в секундах.
Флаг - показывать ось	Задаёт необходимость отображения оси времени на поле графика.
Ось значений	
Поле ввода минимум	Задаёт значение нижней границы диапазона шкалы значений.
Поле ввода максимум	Задаёт значение верхней границы диапазона шкалы значений.
Поле выбора автоподгонка	Задаёт вид отображения диапазона шкалы по оси значений. <i>Нет</i> означает, что диапазон шкалы соответствует заданным границам. <i>В окне</i> означает, что диапазон шкалы автоматически выбирается от минимума до максимума значения параметра на видимом в настоящий момент в окне отрезке графика. <i>За весь период</i> означает, что диапазон автоматически выбирается от минимума до максимума значения параметра за все время отображения графика на мнемосхеме (с момента последнего открытия мнемосхемы)
Флаг Показывать ось	Задаёт необходимость отображения оси значений на поле графика.

На закладке **Параметры** задаются настройки отображения параметров ВФБ График



Элемент диалога	Описание
Список отображаемых параметров	Флажками задаются те параметры, которые должны отображаться на графике, название параметра может быть изменено.
Поле ввода - <i>Ед. изм.</i>	Единица измерения наследуется из свойств переменной дерева объекта и может быть изменена, при отсутствии текста единица измерения не пишется.
Поле ввода - <i>Тип</i>	Тип значения наследуется из свойств переменной дерева объекта и может быть изменен, при этом будет производиться необходимое пре-

Элемент диалога	Описание
Кнопка - <i>Цвет</i>	образование значения. Задаёт цвет, которым будет отображаться график выбранного в списке параметра.
Выпадающий список – <i>Точность</i>	Точность представления числа (число знаков после запятой) наследуется из свойств переменной дерева объекта и может быть изменена.
Поле ввода - <i>Номер ступени</i>	Используется для разнесения по вертикали графиков нескольких дискретных параметров (чтобы избежать наложение их друг на друга), задаёт ступень параметра.
Кнопка - <i>Вернуть шкалу</i>	Восстанавливает свойства унаследованные из дерева объекта.

❖ **Внимание!** При привязке переменных к параметрам графика в дереве объекта необходимо начинать с первого параметра.



Индикатор значения

Индикатор значения предназначен для отображения аналогового значения переменной в виде вертикального или горизонтального столбика. В поле индикатора может отображаться градуированная шкала, поле цифрового значения переменной, единица измерения.

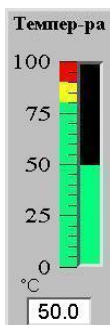
Источником данных для *Индикатора* может быть:

- Любая аналоговая переменная MasterSCADA, за исключением OPC переменной для записи и Входа функционального блока;
- Вход мнемосхемы, связанный с переменной MasterSCADA, либо с локальным выходом мнемосхемы.

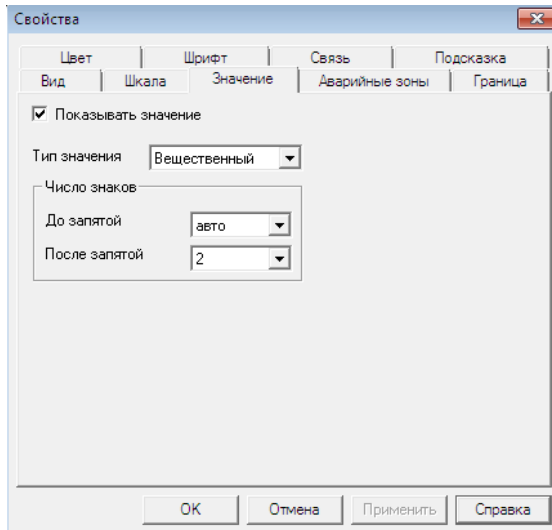
Существует два основных способа привязки *Индикатора* к источнику данных, позволяющих отслеживать значение некоторой переменной в мнемосхеме:

- Перетащить интересующую аналоговую переменную (за исключением *Команды*) в мнемосхему правой кнопкой мыши, и в раскрывающемся контекстном меню выбрать *Индикатор*. При работе графического редактора в составе пакета программ MasterSCADA, этот способ является наиболее предпочтительным.
- Создать *Индикатор* в мнемосхеме, воспользовавшись палитрой элементов графического редактора. Динамизировать свойство *Значение* от входа мнемосхемы, связанного либо с переменной, либо с локальным выходом мнемосхемы.

При вставке в мнемосхему индикатор имеет некоторый стандартный исходный размер. При изменении размера индикатор сохраняет пропорции, поэтому обычно с границей индикатора совпадают только вертикальные или только горизонтальные стороны рамки выделения активного элемента.



На закладке *Значение* настраивается внешний вид параметра *Индикатора*:

**Элемент диалога****Описание**

Флаг - *Показывать значение*

Задаёт необходимость отображения цифрового значения параметра.

Выпадающий список - *Тип значения*

Тип значения наследуется из свойств переменной дерева объекта и может быть изменен.

Выпадающий список - *До запятой*

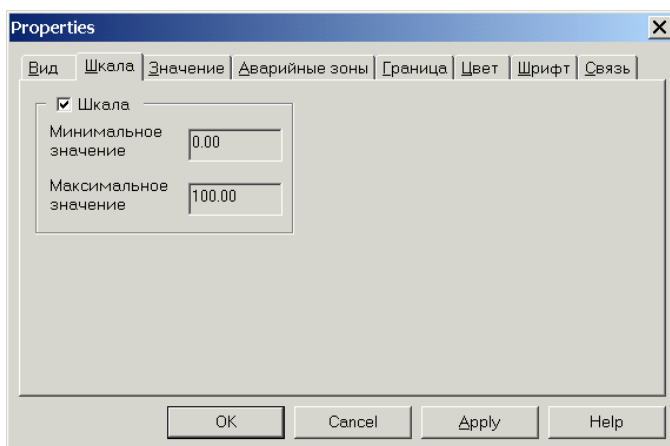
Задаёт число знаков значения до запятой; при превышении этого числа знаков в поле будут показаны символы ошибки #####.

Выпадающий список - *После запятой*

Число знаков значения после запятой наследуется из свойств переменной дерева объекта и может быть изменено.

- ❖ **Важно!** При отсутствии связи элемента управления с переменной, а также при работе графического редактора вне среды пакета программ MasterSCADA, наследования свойств переменной не происходит. В этом случае все свойства элемента настраиваются непосредственно на его страницах свойств. Для того, чтобы установить связь некоторой переменной с элементом управления, необходимо выделить нужный элемент в мнемосхеме, перетащить переменную из дерева объектов в среднюю часть панели настройки и выполнить команду Сервис-Обновить связи.

На закладке *Шкала* отображаются настройки шкалы элемента управления

**Элемент диалога****Описание**

Флаг - *Шкала*

Задаёт необходимость отображения градуированной шкалы рядом со столбцом значения.

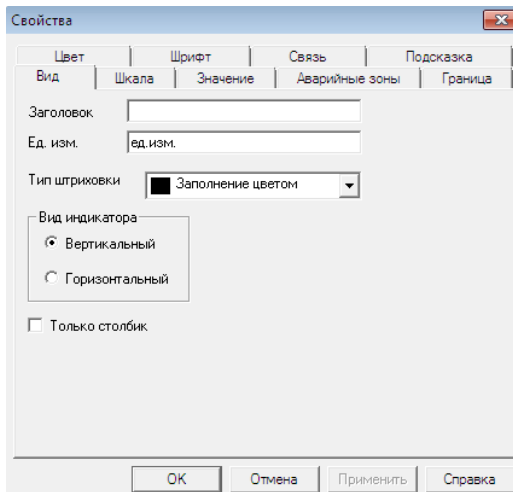
Поле ввода - *Минимальное значение*

Минимальное значение диапазона шкалы наследуется из свойств переменной дерева объектов и не может быть изменено.

Поле ввода - *Максимальное значение*

Максимальное значение диапазона шкалы наследуется из свойств переменной дерева объектов и не может быть изменено.

На закладке *Вид* настраивается внешний вид элемента управления *Индикатор*.



Элемент диалога

Описание

Поле ввода - *Заголовок*

Имя наследуется от имени переменной в дереве объектов, новое имя может быть задано в поле ввода, при отсутствии текста индикатор заголовка не имеет.

Поле ввода *названия ед. измерения*

Ед. измерения наследуется из свойств переменной дерева объекта и может быть изменена, при отсутствии текста единица измерения не отображается.

Выпадающий список - *Тип штриховки*

Задаёт тип заполнения поверхности элемента управления.

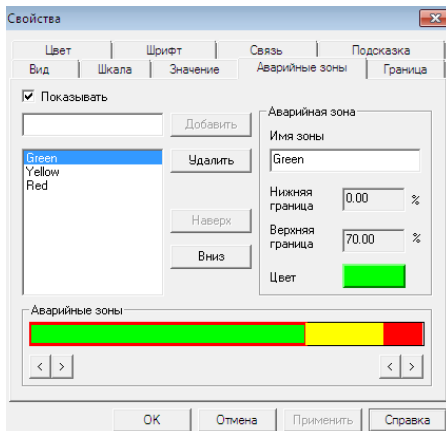
Поле выбора - *Вид индикатора*

Задаёт вертикальное или горизонтальное расположение столбика индикатора.

Флаг - *Только столбик*

Задаёт отображение *Индикатора*, как столбика (без заголовка, шкалы, единиц измерения и значения).

На закладке *Аварийные зоны* показываются все зоны сигнализации элемента управления. Эта закладка доступна для *Стрелочного прибора* и *Индикатора*.



Эта закладка может открываться в двух режимах:

- Только для просмотра - если данный элемент имеет связь с переменной в дереве объектов. В этом случае настройки аварийных зон наследуются из настроек самой переменной;
- С возможностью изменения - если данный элемент не имеет связи с переменной дерева объектов. В этом случае настройки аварийных зон производятся на странице свойств.

Элемент диалога

Описание

Флаг - *Показывать*

Задаёт необходимость отображения аварийных зон.

Список аварийных зон

Показывает все существующие зоны сигнализации. Отображает имя выбранной в списке аварийной зоны.

Поле ввода - *Имя зоны*

Нижняя граница аварийной зоны в процентах наследуется из свойств.

Поле ввода - *Нижняя граница*

Верхняя граница аварийной зоны в процентах наследуется из свойств.

Поле ввода - *Верхняя граница*

Кнопка - *Цвет*

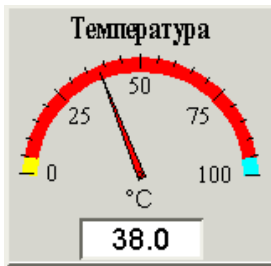
Цвет аварийной зоны.

Кнопки

Позволяют настроить границы аварийных зон. Кнопки, расположенные слева настраивают левую границу выделенной зоны, расположенные справа - правую.

Стрелочный индикатор

Стрелочный прибор предназначен для отображения значения аналоговой переменной. В поле стрелочного прибора может отображаться градуированная шкала, поле цифрового значения переменной, единица измерения.



Источником данных для *Стрелочного прибора* может быть:

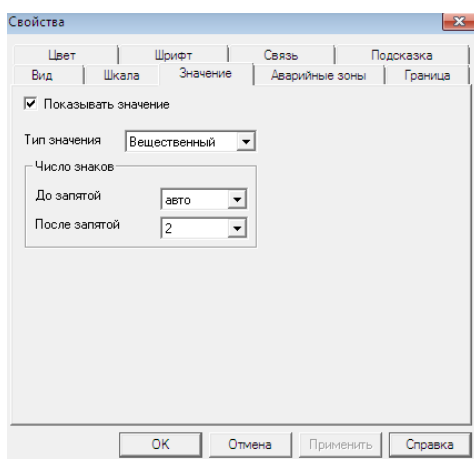
- Любая аналоговая переменная MasterSCADA, за исключением OPC переменной для записи и Входа функционального блока;
- Вход мнемосхемы, связанный с переменной MasterSCADA, либо с локальным выходом мнемосхемы.

Существует два основных способа привязки *Стрелочного прибора* к источнику данных, позволяющих отслеживать значение некоторой переменной в мнемосхеме:

- Перетащить интересующую аналоговую переменную (за исключением *Команды*) в мнемосхему правой кнопкой мыши и в открывшемся контекстном меню выбрать *Стрелочный прибор*. При работе графического редактора в составе пакета программ MasterSCADA этот способ является наиболее предпочтительным.
- Создать *Стрелочный прибор* в мнемосхеме, воспользовавшись палитрой элементов графического редактора. Динамизировать свойство *Значение* от входа мнемосхемы, связанного либо с переменной MasterSCADA, либо с локальным выходом мнемосхемы.

При вставке в мнемосхему *Стрелочный прибор* имеет некоторый стандартный исходный размер. При изменении размера элемента управления сохраняет пропорции, поэтому обычно с его границей совпадают только вертикальные или только горизонтальные стороны рамки выделения активного элемента. Свойства стрелочного прибора (вид, шкала, значение, аварийные зоны, граница, цвет, шрифт, связь, подсказка) настраиваются аналогично свойствам, указанным в предыдущих разделах.

На закладке *Значение* настраивается внешний вид параметра *Стрелочного прибора*:

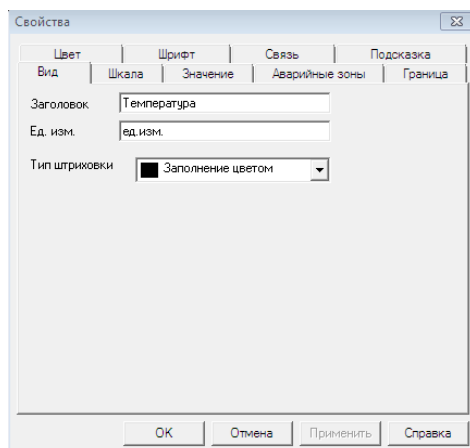


Элемент диалога

Описание

Флаг - <i>Показывать значение</i>	Задаёт необходимость отображения цифрового значения параметра.
Выпадающий список - <i>Тип значения</i>	Тип значения наследуется из свойств переменной дерева объекта и может быть изменен.
Выпадающий список - <i>До запятой</i>	Задаёт число знаков значения до запятой; при превышении этого числа знаков в поле будут показаны символы ошибки #####.###.
Выпадающий список - <i>После запятой</i>	Число знаков значения после запятой наследуется из свойств переменной дерева объекта и может быть изменено.

На закладке *Вид* настраивается внешний вид *Стрелочного прибора*.



Элемент диалога

Описание

Поле ввода - *Заголовок*

Имя наследуется от имени переменной в дереве объектов, новое имя может быть задано в поле ввода, при отсутствии текста *Стрелочный прибор* заголовка не имеет.

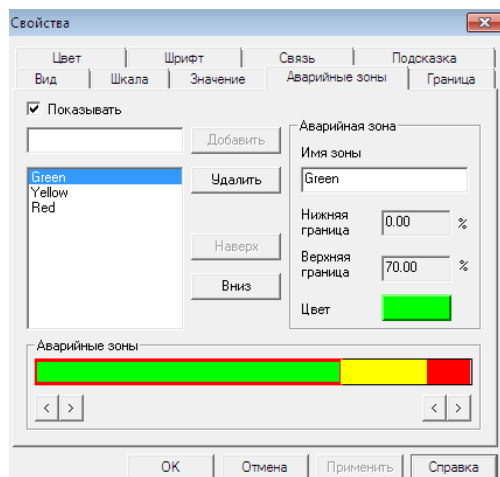
Поле ввода названия ед. измерения

Ед. измерения наследуется из свойств переменной и может быть изменена, при отсутствии текста единица измерения не отображается.

Выпадающий список - *Тип штриховки*

Задаёт тип заполнения штриховки поверхности *Стрелочного прибора*.

На закладке *Аварийные зоны* показываются все зоны сигнализации элемента управления. Эта закладка доступна для *Стрелочного прибора* и *Индикатора*.



Эта закладка может открываться в двух режимах:

- Только для просмотра - если данный элемент имеет связь с переменной в дереве объектов. В этом случае настройки аварийных зон наследуются из настроек самой переменной;
- С возможностью изменения - если данный элемент не имеет связи с переменной дерева объектов. В этом случае настройки аварийных зон производятся на странице свойств.

Элемент диалога

Описание

Флаг - *Показывать*

Задаёт необходимость отображения аварийных зон.

Список аварийных зон

Показывает все существующие зоны сигнализации.

Поле ввода - *Имя зоны*

Отображает имя выбранной в списке аварийной зоны.

Поле ввода - *Нижняя граница*

Нижняя граница аварийной зоны в процентах наследуется из свойств.

Поле ввода - *Верхняя граница*

Верхняя граница аварийной зоны в процентах наследуется из свойств.

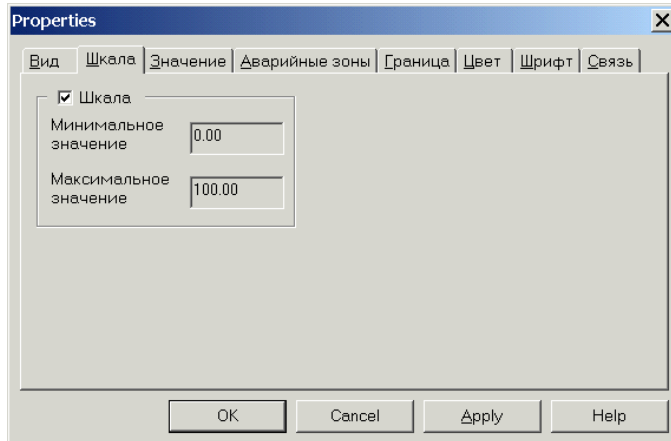
Кнопка - *Цвет*

Цвет аварийной зоны.

Кнопки 

Позволяют настроить границы аварийных зон. Кнопки, расположенные слева настраивают левую границу выделенной зоны, расположенные справа – правую.

На закладке *Шкала* отображаются настройки шкалы элемента управления



Элементы диалога	Описание
Флаг - <i>Шкала</i>	Задает необходимость отображения градуированной шкалы рядом со столбцом значения.
Поле ввода - <i>Минимальное значение</i>	Минимальное значение диапазона шкалы наследуется из свойств переменной дерева объектов и не может быть изменено.
Поле ввода - <i>Максимальное значение</i>	Максимальное значение диапазона шкалы наследуется из свойств переменной дерева объектов и не может быть изменено.

Задатчик

Основное назначение **Задатчика** - быть источником данных для аналоговой Команды в дереве объектов. Кроме того, **Задатчик** может быть использован и для изменения свойств других элементов мнемосхем. В поле **Задатчика** может отображаться градуированная шкала, поле цифрового значения переменной, единица измерения.

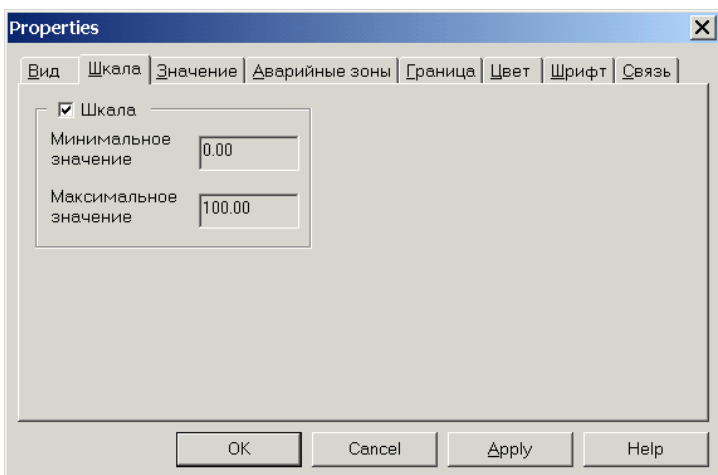


Для того, чтобы управлять выходом *Команды* с помощью *Задатчика*, нужно сделать следующее:

- Перетащить нужную *аналоговую Команду* в мнемосхему правой кнопкой мыши, и в раскрывающемся контекстном меню выбрать *Задатчик*. При работе графического редактора в составе пакета программ MasterSCADA, этот способ является наиболее предпочтительным.
- Создать *Задатчик* в мнемосхеме, воспользовавшись палитрой элементов графического редактора. Динамизировать свойство задатчика *Значение* по выходу, связанному с *Командой*.

При вставке в мнемосхему задатчик имеет некоторый стандартный исходный размер. При изменении размера задатчика сохраняет пропорции, поэтому обычно с границей задатчика совпадают только вертикальные или только горизонтальные стороны рамки выделения активного элемента.

На закладке *Шкала* отображаются настройки шкалы *Задатчика*.



Элемент диалога

Описание

Флаг - *Шкала*

Задаёт необходимость отображения градуированной шкалы рядом со столбцом значения.

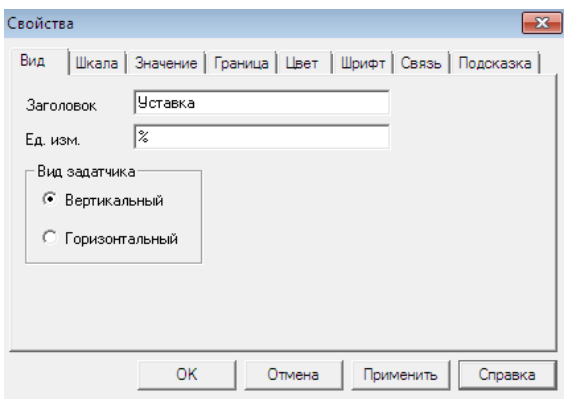
Поле ввода - *Минимальное значение*

Минимальное значение диапазона шкалы наследуется из свойств переменной дерева объекта и не может быть изменено.

Поле ввода - *Максимальное значение*

Максимальное значение диапазона шкалы наследуется из свойств переменной дерева объекта и не может быть изменено.

На закладке *Вид* настраивается внешний вид элемента управления *Задатчик*.



Элемент диалога

Описание

Поле ввода - *Заголовок*

Имя наследуется от имени переменной в дереве объектов, новое имя может быть задано в поле ввода, при отсутствии текста элемент управления заголовка не имеет.

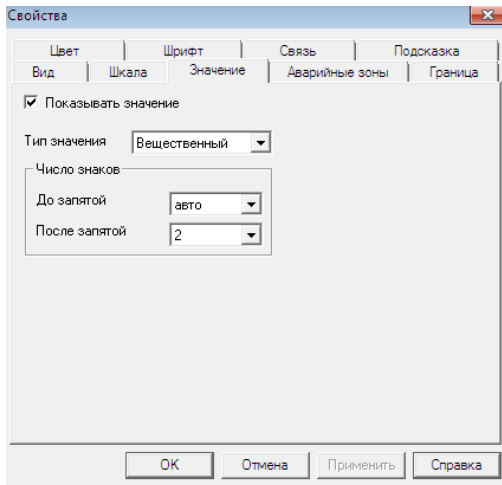
Поле ввода названия ед. измерения

Ед. измерения наследуется из свойств переменной дерева объектов и может быть изменена, при отсутствии текста единица измерения не отображается.

Поле выбора - *Вид задатчика*

Задаёт вертикальное или горизонтальное расположение ползунка задатчика.

На закладке *Значение* настраивается внешний вид параметра *Задатчика*.



Элемент диалога

Описание

Флаг - <i>Показывать значение</i>	Задаёт необходимость отображения цифрового значения параметра.
Выпадающий список - <i>Тип значения</i>	Тип значения наследуется из свойств переменной дерева объекта и может быть изменен.
Выпадающий список - <i>До запятой</i>	Задаёт число знаков значения до запятой; при превышении этого числа знаков в поле будут показаны символы ошибки #####.###.
Выпадающий список - <i>После запятой</i>	Число знаков значения после запятой наследуется из свойств переменной дерева объекта и может быть изменено.

Значение

Элемент управления **Значение** предназначен как для отображения значения переменной, так и для ввода данных. В поле значения, кроме цифрового значения переменной, может отображаться имя переменной и единица измерения.

Тем-ра воды 76.9 °C

Существует два основных способа использования элемента управления **Значение**:

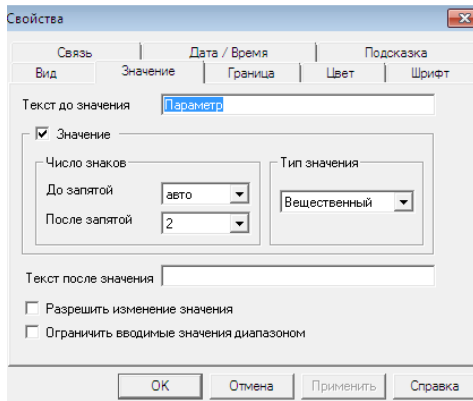
- Перетащить интересующую переменную в мнемосхему левой кнопкой мыши. В этом случае настройка основных параметров **Значения** производится автоматически, в соответствии с типом и атрибутами связанной переменной:

Переменная	Тип Значения (элемент мнемосхемы)
<i>Команда</i>	Редактируемое значение - в режиме исполнения, по двойному щелчку на значении открывается диалоговое окно, позволяющее осуществить ввод данных в Команду.
<i>Значение, Событие, Расчет, OPC переменная для чтения, Выход ФБ</i>	Элемент мнемосхемы Значение предназначен только для просмотра данных выхода соответствующей переменной.

Кроме того, **Значение** наследует от связанной переменной диапазон шкал, единицу измерения и настройки аварийных зон. Таким образом, останется настроить только внешний вид элемента управления;

- Вставить **Значение** в мнемосхему из палитры элементов графического редактора. Динамизировать свойство **Значение** элемента управления традиционным способом от входа мнемосхемы, связанного с нужной переменной (для просмотра данных), либо по выходу мнемосхемы, связанному либо с **Командой**, либо с локальным входом мнемосхемы (для управления). Очевидно, что данный подход является более трудоемким, в связи с необходимостью ручной настройки динамизации, а также свойств **Значения**.

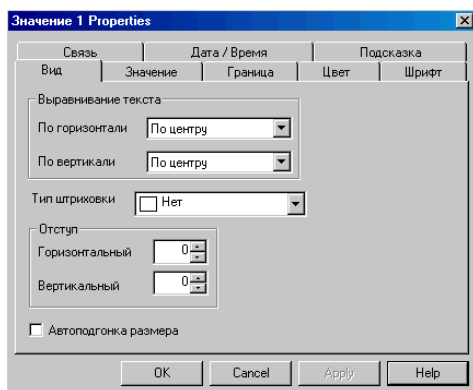
На закладке **Значение** настраивается внешний вид параметра элемента управления **Значение**:



Элемент диалога	Описание
Поле ввода - <i>Текст до значения</i>	Наследуется от имени переменной в дереве объекта, новое имя может быть задано в поле ввода (если в поле ввода текст отсутствует, то Значение будет начинаться с цифрового значения переменной).
Флаг - <i>Значение</i>	Задаёт необходимость отображения цифрового значения параметра.
Выпадающий список - <i>До запятой</i>	Задаёт число знаков значения до запятой; при превышении этого числа знаков в поле будут показаны символы ошибки #####.
Выпадающий список - <i>После запятой</i>	Число знаков значения после запятой наследуется из свойств переменной дерева объекта и может быть изменено.
Выпадающий список - <i>Тип значения</i>	Тип значения наследуется из свойств переменной дерева объекта и может быть изменен.
Поле ввода - <i>Текст после значения</i>	По умолчанию задаёт ед. измерения, которая наследуется из свойств переменной дерева объекта и может быть изменена (при отсутствии текста единица измерения не отображается).
Флаг - <i>Разрешить изменение значения</i>	Задаёт возможность изменения значения переменной с мнемосхемы (в режиме исполнения при двойном щелчке по изображению значения открывается диалог задания значения переменной).
Флаг - <i>Ограничить вводимые значения диапазоном</i>	Задаёт возможность ограничить вводимые значения диапазоном. При условии, если команде назначена <i>Шкала</i> .

❖ **Важно!** При отсутствии связи элемента управления с переменной, а также при работе графического редактора вне среды пакета программ MasterSCADA, наследования свойств переменной не происходит. В этом случае все свойства элемента настраиваются непосредственно на его страницах свойств. Для того, чтобы установить связь некоторой переменной с элементом управления, необходимо выделить нужный элемент в мнемосхеме, перетащить переменную из дерева объектов в среднюю часть панели настройки и выполнить команду Сервис-Обновить связи.

На закладке *Вид* настраивается внешний вид элемента управления *Значение*.

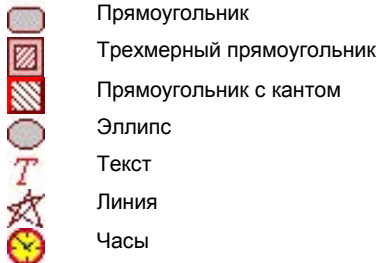


Элемент диалога	Описание
Выпадающий список - <i>По горизонтали</i>	Задаёт способ выравнивания текста по горизонтали.
Выпадающий список - <i>По вертикали</i>	Задаёт способ выравнивания текста по вертикали.
Выпадающий список - <i>Тип штриховки</i>	Задаёт тип заполнения поля <i>Значения</i> .
Поле ввода - <i>Горизонтальный отступ</i>	Задаёт отступ текста по горизонтали.
Поле ввода - <i>Вертикальный отступ</i>	Задаёт отступ текста по вертикали.
Флаг - <i>Автоподгонка размера</i>	Задаёт автоподгонку размера поля значения по ширине данных (имя, значение и единица измерения).

Графические примитивы

Графические примитивы используются для создания объектов мнемосхем и отображают эти объекты.

Графические примитивы бывают следующих типов:



Каждый графический примитив имеет собственные свойства, доступ к которым можно получить, открыв диалог *Свойства* или используя панели настройки, однако, графические примитивы не имеют своих собственных событий.

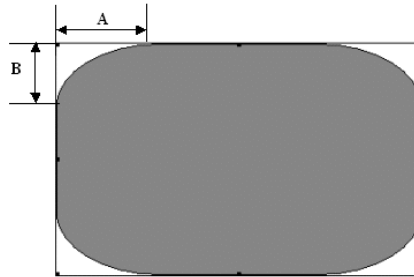
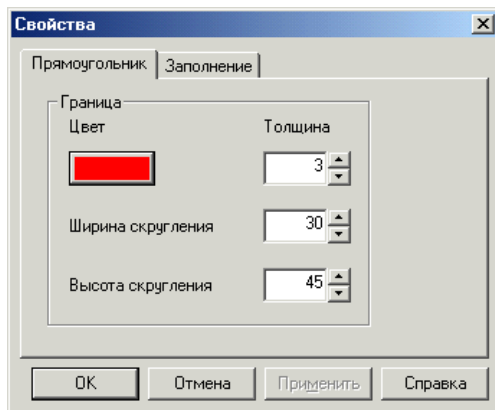
Добавить графические примитивы на мнемосхему можно любым из двух способов: с помощью *Панели настройки* или *Панели элементов*.

Прямоугольник

Скругленный прямоугольник представляет собой прямоугольник с закругляющимися углами.

Свойства скругленного прямоугольника настраиваются в диалоге *Свойства*, вызываемом из пункта контекстного меню. Диалог свойства состоит из двух закладок: *Прямоугольник* и *Заполнение*.

На закладке *Свойства* настраивается внешний вид *Прямоугольника*.



На данной закладке настраиваются следующие свойства:

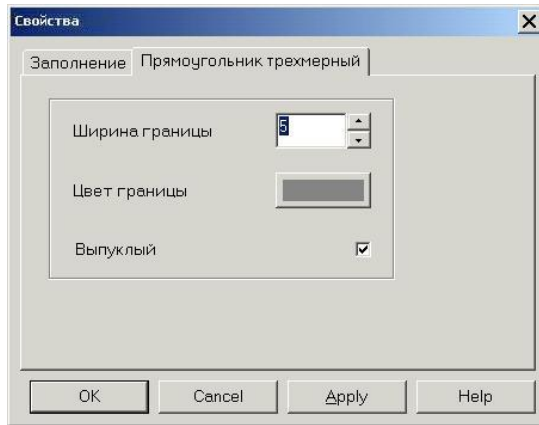
- *Цвет* - цвет рамки (границы) прямоугольника. Кнопка рядом с надписью *Цвет* окрашивается в текущий цвет границы прямоугольника. Изменение цвета границы происходит в диалоге *Цвет*, который вызывается при нажатии на эту кнопку.
- *Толщина* - толщина рамки (границы) *Прямоугольника*.
- *Ширина скругления* - величина, равная длине отрезка *A*.
- *Высота скругления* - длина отрезка *B*.

Трёхмерный прямоугольник

Трёхмерный прямоугольник - это выпуклый/вдавленный прямоугольник, т.е. прямоугольник в виде кнопки.

Диалог *Свойства* состоит из двух закладок: *Прямоугольник в виде кнопки* и *Заполнение*.

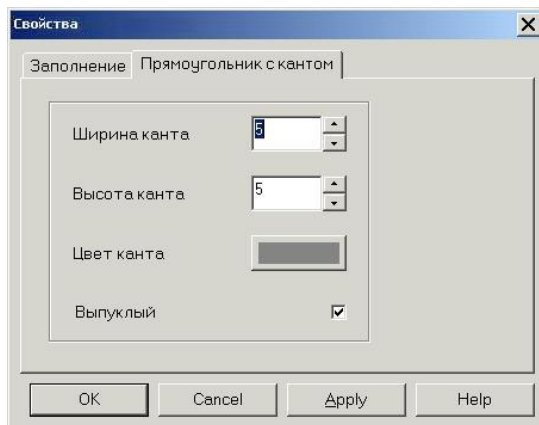
На закладке *Прямоугольник трёхмерный* настраивается внешний вид *Прямоугольника*.



Ширина границы - ширина рамки (границы) прямоугольника. *Цвет границы* - цвет рамки (границы) прямоугольника. *Выпуклый* - определяет вид прямоугольника: выпуклый или вдавленный.

Прямоугольник с кантом

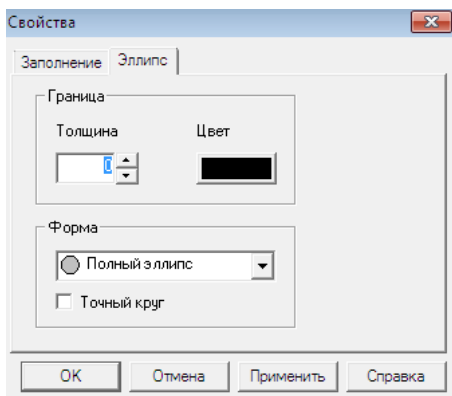
Прямоугольник с кантом - это прямоугольник, обрамленный выпуклой/вдавленной границей (кантом). Свойства *Прямоугольника* с кантом настраиваются в диалоге *Свойства*, вызываемом из пункта контекстного меню. Диалог *Свойства* состоит из двух закладок: *Прямоугольник с кантом* и *Заполнение*.



Ширина канта - ширина рамки (границы) прямоугольника. *Высота канта* - ширина рамки (границы) прямоугольника. *Цвет канта* - цвет рамки (границы) прямоугольника. *Выпуклый* - определяет вид прямоугольника: выпуклый или вдавленный.

Эллипс

Эллипс - геометрическая фигура эллипс с заполнением. Свойства *Эллипса* настраиваются в диалоге *Свойства*, вызываемом из пункта контекстного меню. Диалог *Свойства* состоит из двух закладок: *Эллипс* и *Заполнение*. На закладке *Эллипс* настраивается толщина и цвет границы, а также тип фигуры.



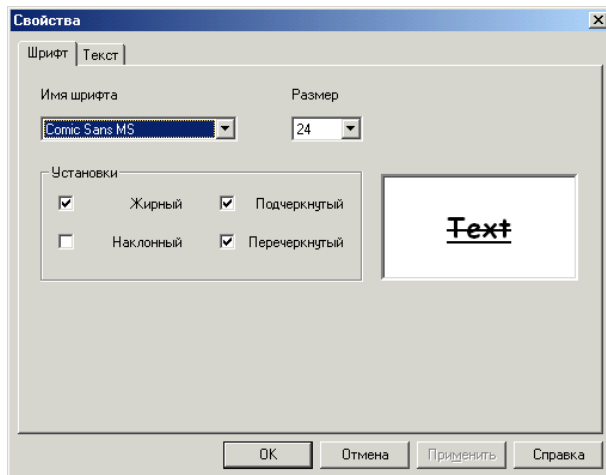
Ширина - ширина рамки (границы) эллипса. *Цвет* - цвет рамки (границы) эллипса. Кнопка рядом с надписью *Цвет* окрашивается в текущий цвет границы эллипса. Изменение цвета границы происходит в диалоге *Цвет*, который вызывается при нажатии на эту кнопку.

Форма - тип *Эллипса*:

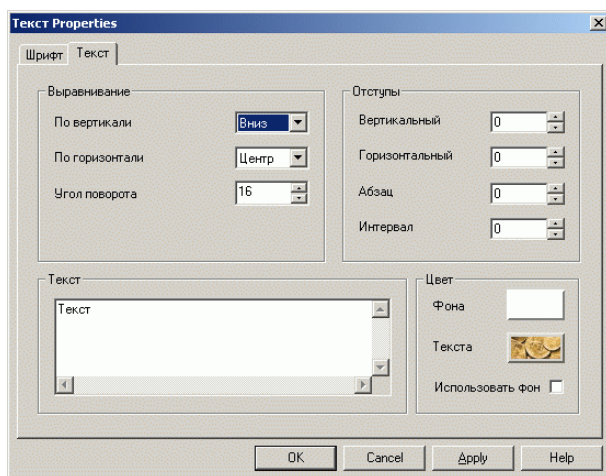
- Полный эллипс;
- Верхняя половина;
- Нижняя половина;
- Левая половина;
- Правая половина;
- Левая верхняя четверть;
- Правая верхняя четверть;
- Левая нижняя четверть;
- Правая нижняя четверть.

Текст

Элемент *Текст* служит для отображения текста, причем текст может содержать несколько строк. Ниже описаны свойства *Текста* в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Диалог *Свойства* состоит из двух закладок: *Шрифт* и *Текст*. На закладке "Шрифт" настраиваются атрибуты шрифта: название шрифта, его размер, стиль (жирный, подчеркнутый, наклонный и перечеркнутый).

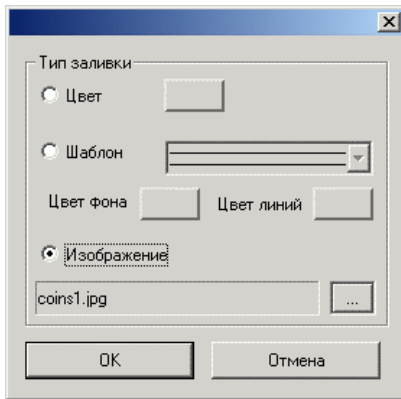


Закладка *Текст* служит для ввода текста, указания выравнивания, цвета фона и текста.



В области *Выравнивание* - выравнивание текста по вертикали (вверх, по центру, вниз) и по горизонтали (влево, вправо, по центру), определяет угол поворота текста. Поле *Текст* служит для ввода самого текста. При вводе нескольких строк, перевод строки осуществляется нажатием клавиш *Ctrl + Enter*. Область *Цвет* определяет цвет фона и текста. Кнопки, расположенная в группе *Цвет* окрашиваются в текущий цвет фона и текста. Изменение цвета фона в диалоге *Цвет*, который вызывается при нажатии на эту кнопку. Если отображение фона не требуется снимите флажок *Использовать фон*. В этом случае фон - *прозрачный*. При

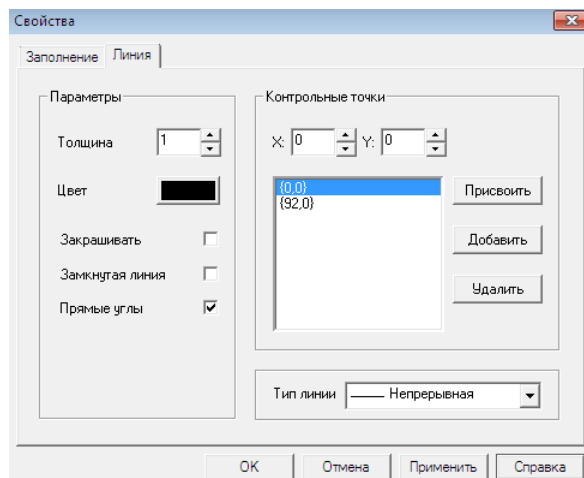
установке флажка *Использовать фон* текст отображается на фоне цветом, указанном в группе *Цвет*. При нажатии на кнопку *Цвет текста*, открывается диалоговое окно для установки типа заливки.



При выборе кнопки *Цвет*, становится активной кнопка, при нажатии на которую открывается стандартное диалоговое окно для определения цвета. При выборе кнопки *Шаблон*, устанавливаются стандартные виды текстур, которые можно определить в выпадающем списке. Для них можно установить цвета линий и фона, нажав на соответствующие кнопки. При выборе кнопки *Изображение*, открывается окно, в котором можно выбрать графический файл, который будет использоваться в качестве текстуры для текста.

Линия

Линия - позволяет рисовать линии произвольной толщины и стиля. Ниже описаны свойства Линии в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Диалог *Свойства* состоит из двух закладок: *Линия* и *Заполнение*. На закладке *Линия* настраивается внешний вид Линии.



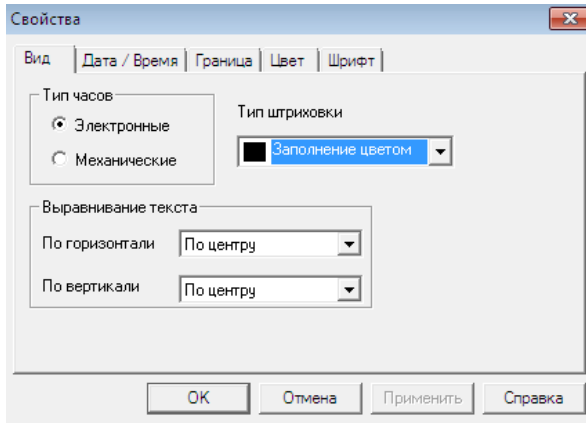
Толщина - толщина линии. *Цвет* - цвет линии. *Закрашивать* - при установке флага, площадь очерченная *Линией* закрашивается выбранным цветом. *Замкнутая линия* - при установке этого флага линия становится замкнутой, т.е. первая точка линии соединяется с последней. *Прямые углы* - при установке флажка, все изгибы линии рисуются под прямым углом. *Контрольные точки* - добавление и удаление точек линии и редактирование их координат. *Тип линии* - тип линии (справа показывается как линия будет отображаться на экране):

- Непрерывная;
- Пунктир;
- Точки;
- Штрих-пунктир;
- Пунктир 1;
- Пунктир 2;
- Пунктир 3;

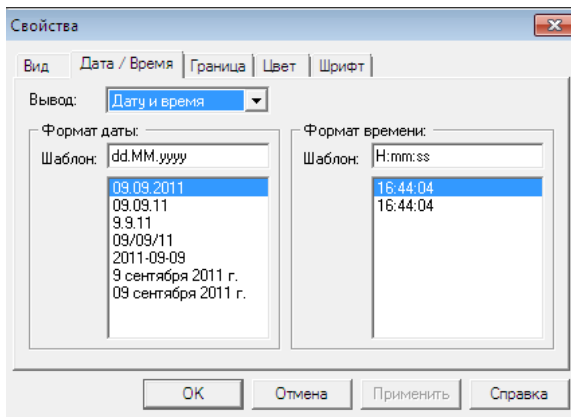
Часы

Элемент *Часы* служит для отображения даты и времени. Ниже описаны свойства *Часов* в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Диалог *Свойства* состоит из пяти закладок: *Вид*, *Дата/Время*, *Граница*, *Цвет* и *Шрифт*.

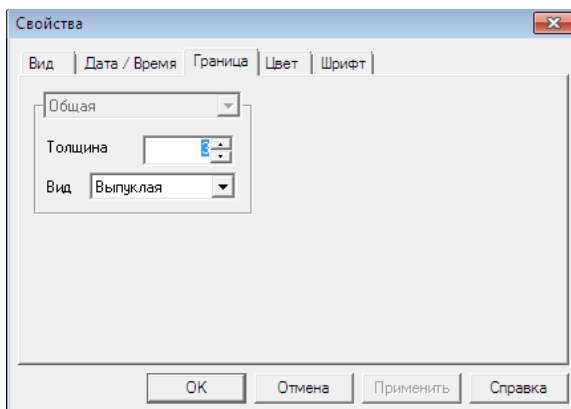
На закладке *Вид* настраивается внешний вид часов: тип часов (электронные или механические), тип заполнения элемента, выравнивание текста по вертикали и горизонтали.



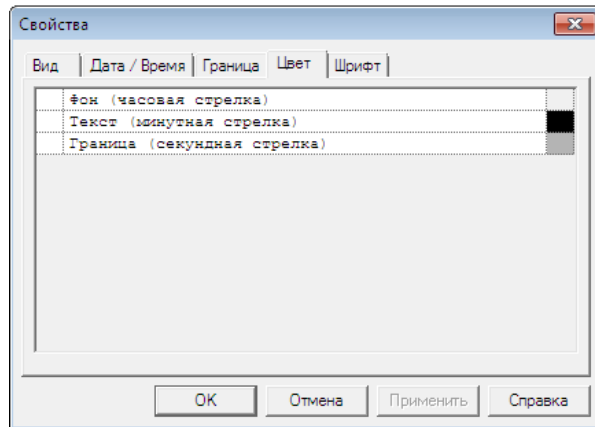
На закладке *Дата/Время* настраиваются атрибуты часов - вывод элемента (что будет выводиться на элемент: дата или время, или и первое и второе), формат даты и формат времени.



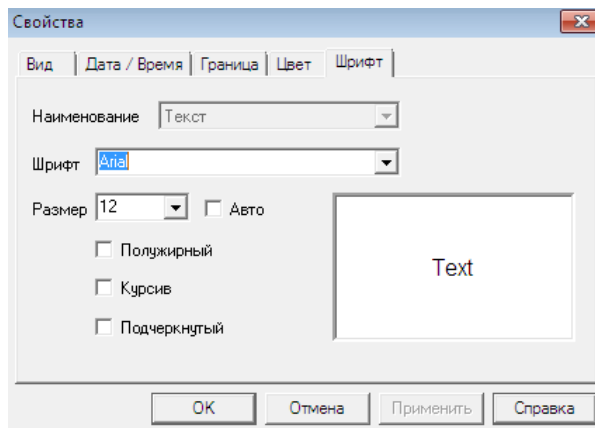
На закладке *Граница* настраиваются атрибуты границы элемента: толщина границы и ее вид (выпуклая, плоская и вогнутая).



На закладке *Цвет* настраиваются цвета атрибутов часов: фона, текста и границы (для электронных) и часовых стрелок (для механических).

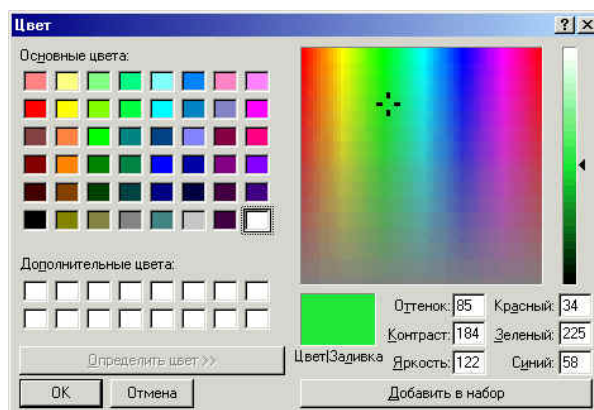


На закладке *Шрифт* настраиваются атрибуты шрифта: название шрифта, его размер, стиль (полужирный, подчеркнутый, курсив).



Общие свойства объектов

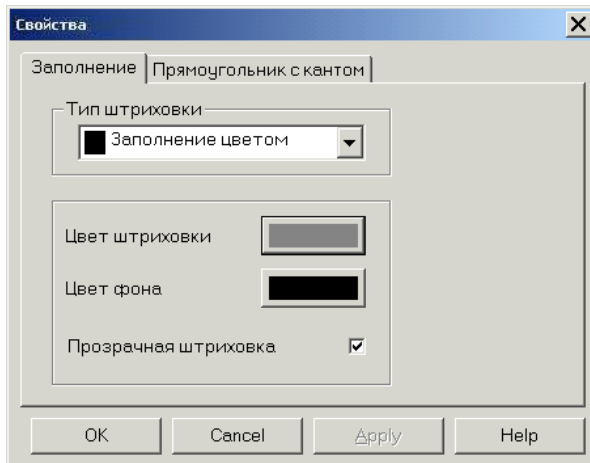
Стандартное диалоговое окно *Цвет* служит для выбора дополнительных цветов. В этом диалоге можно выбрать цвет из основной палитры или создать новый цвет.



Создать новый цвет можно щелкнув в большой правый квадрат. В месте щелчка появится перекрестие и область *Цвет|Заливка* окрасится в выбранный цвет. Оттенок цвета можно изменить при помощи ползунка, расположенного справа от большого квадрата. Определить цвет можно также вводом в поля *Оттенок*, *Контраст*, *Яркость* чисел от 0 до 240 и в поля *Красный*, *Зеленый*, *Синий* чисел от 0 до 255. Кнопка *Добавить в набор* добавляет созданный цвет в дополнительные цвета основной палитры.

Закладка *Заполнение* служит для указания типа заполнения (штриховки) элемента. *Цвет штриховки* - цвет рисунка штриховки. Кнопка, расположенная в группе *Цвет штриховки* окрашивается в текущий цвет. *Цвет фона* - цвет заполнения элемента. Кнопка, расположенная в группе *Цвет фона* окрашивается в текущий цвет. Изменение цвета элемента происходит в диалоге *Цвет*, который вызывается при нажатии на эту кнопку. *Прозрачная*

штриховка - задает прозрачность элемента. При отсутствии флажка, элемент имеет цвет заполнения.



Тип штриховки - тип заполнения элемента (слева от вида штриховки показывается как она будет отображаться на экране):

- Заполнение цветом;
- Горизонтальная;
- Вертикальная;
- Левая диагональная;
- Правая диагональная;
- Прямая сетка;
- Диагональная сетка.

Элементы диалога

Управляющие элементы позволяют пользователю эффективнее и удобнее работать с мнемосхемой в режиме реального времени. *Управляющие элементы* — это графические объекты, размещаемые на мнемосхеме для отображения или ввода данных, выполнения действий или облегчения восприятия мнемосхемы. Эти объекты включают списки, группы переключателей, кнопки и т.п. Они позволяют пользователю выбирать различные параметры или выполнять действия (события), служащие для автоматизации задач. Каждый управляющий элемент имеет как общие (настройка строк и шрифта), так и специфические (вид элемента, параметры строк у блока параметров) свойства, доступ к которым можно получить, открыв диалог "Свойства" или используя панель "Свойства".

Элемент Кнопка

Кнопки служат для выполнения определенного действия (вызов мнемосхемы, присвоение значения и т.д.). Как и у любого элемента у кнопки есть события и свойства.

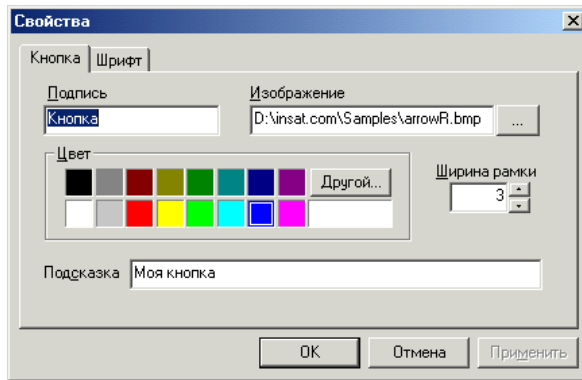


Помимо стандартных событий этот элемент имеет еще два своих события: нажатие и отпускание *Кнопки*:

- Событие *Нажатие кнопки* происходит при нажатии на кнопку левой клавишей мышью, т.е. оно ничем не отличается от стандартного общего события по левой кнопке;
- Событие *Отпускание кнопки* происходит, когда нажатая левая кнопка мыши отпускается.

Ниже описаны свойства кнопки в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели *Свойства*. Диалог *Свойства* контекстного меню состоит из двух закладок: *Кнопка* и *Шрифт*.

На закладке *Кнопка* настраивается внешний вид *Кнопки*.



Подпись - надпись на *Кнопке*. *Изображение* - рисунок, помещаемый на *Кнопку* из файла *.BMP*. Рисунок помещается сверху подписи и является необязательным, т.е. может отсутствовать. *Цвет* - цвет *Кнопки*. Цвет выбирается выбором одного из 16-ти цветов (цветных квадратов). Если этих цветов недостаточно, можно выбрать дополнительные цвета из стандартного диалога *Цвет*, вызываемого по кнопке *Другой*. *Ширина рамки* - ширина рамки, очерчивающей *Кнопку*. Если рамка не нужна поставьте *0*. *Подсказка* - текст подсказки, появляющейся в режиме реального времени при наведении на кнопку указателя мыши.

Элемент Кнопка с фиксацией

Кнопка с фиксацией похожа на обыкновенную *Кнопку*, за исключением того, что при нажатии на нее левой клавишей мыши и отпускании этой клавиши ее положение фиксируется, т.е. она остается нажатой. Если в нажатом состоянии на нее нажать, то она отжимается. Поэтому ее следует использовать для выполнения действий, с двумя положениями (*Вкл./Выкл.*). Как и у любого элемента у кнопки с фиксацией есть события и свойства.

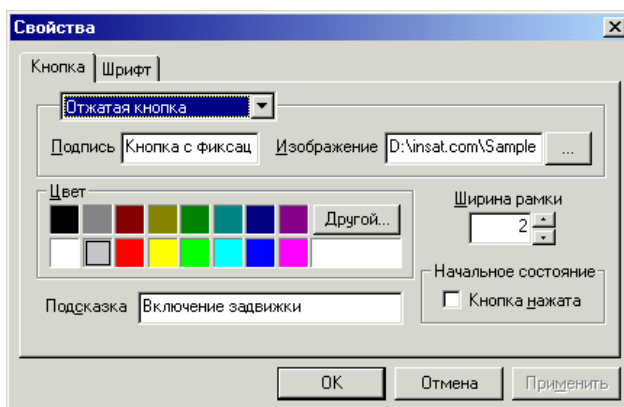


Помимо стандартных событий этот элемент имеет еще два своих события: фиксация и освобождение кнопки.

- Событие *Фиксация кнопки* происходит при клике (нажатии и отпускании) левой клавиши мыши на кнопку. При этом кнопка переходит в нажатое состояние;
- Событие *Освобождение кнопки* происходит также при клике левой клавиши мыши на кнопку. При этом кнопка переходит в нормальное (ненажатое) состояние.

Ниже описаны свойства *Кнопки с фиксацией* в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели *Свойства*. Диалог *Свойства* контекстного меню состоит из двух закладок: *Кнопка* и *Шрифт*.

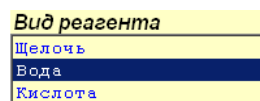
На закладке *Кнопка* настраивается внешний вид *Кнопки с фиксацией* для двух ее состояний: нажатая и отжатая кнопка. Активное состояние (свойства которого настраиваются на этой закладке) устанавливаются в списке, расположенном вверху закладки.



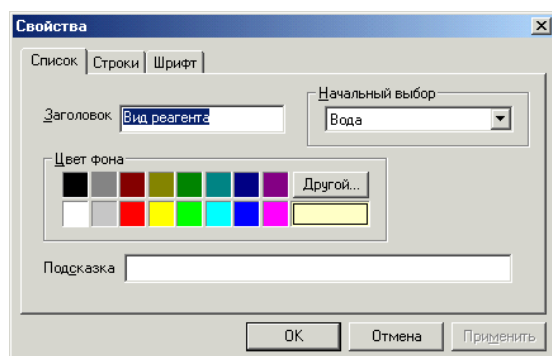
Элементу управления аналогичны элементам *Кнопки*.

Элемент Список

Элемент *Список* представляет собой поле, содержащее список строк. *Списки* используются для выбора определенных строк-параметров. Как и у любого элемента у *Списка* есть события и свойства.



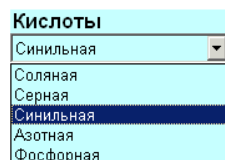
Помимо стандартных событий этот элемент имеет еще одно событие - *Выбор строки*. Данное событие выполняется тогда, когда происходит выбор строки. Выбор строки осуществляется щелчком мыши на нужную строку. Ниже описаны свойства списка в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели *Свойства*. Диалог *Свойства* контекстного меню состоит из трех закладок: *Список*, *Строки* и *Шрифт*. На закладке *Список* настраивается внешний вид *Списка*.




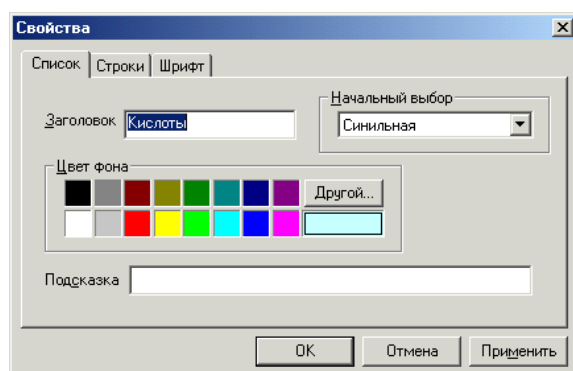
Заголовок - заголовок *Списка*; выводится в верхней части списка. *Начальный выбор* - строка, выбираемая по умолчанию. Остальные элементы управления аналогичны элементам *Кнопки*.

Элемент Выпадающий список

Элемент *Выпадающий список* представляет собой свернутый в одно поле *Список*. После выбора в списке какого-либо элемента, этот элемент остается в текстовом поле списка. Выпадающие списки используются для выбора определенных строк-параметров. Как и у любого элемента у *Выпадающего списка* есть события и свойства.



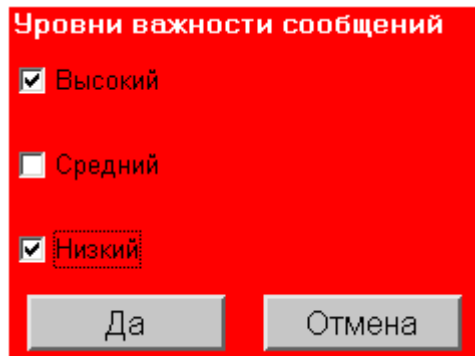
Помимо стандартных событий этот элемент имеет еще одно событие - *Выбор строки*. Данное событие выполняется тогда, когда происходит выбор строки. Для выбора строки откройте список, щелкнув на кнопке  и выберите нужную строку. Ниже описаны свойства выпадающего списка в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели настройки. Диалог *Свойства* контекстного меню состоит из трех закладок: *Список*, *Строки* и *Шрифт*. На закладке *Список* настраивается внешний вид *Выпадающего списка*.



Заголовок - заголовок *Выпадающего списка*, выводится в верхней части списка. *Начальный выбор* - строка, выбираемая по умолчанию. Остальные элементы управления аналогичны элементам *Кнопки*.

Элемент Поля выбора

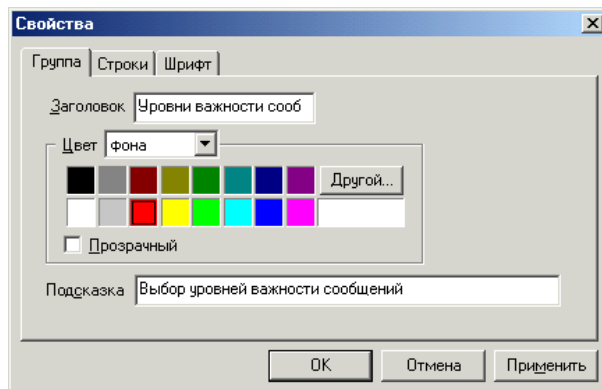
Элемент *Поля выбора* используется для объединения дискретных параметров в группу (блок) и установления значений этих параметров. Как и у любого элемента у блока *Полей выбора* есть события и свойства.



Помимо стандартных событий этот элемент имеет еще одно событие - изменение отметок:

- Событие *Изменение отметок* выполняется когда пользователь изменяет отметки (флажки). Изменение отметок производится установлением/снятием отметок и нажатием кнопки *Да* для подтверждения или кнопки *Отмена* для отмены изменений.

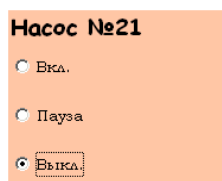
Ниже описаны свойства блока полей выбора в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели *Свойства*. Диалог *Свойства* контекстного меню состоит из трех закладок: *Группа*, *Строки* и *Шрифт*. На закладке *Группа* настраивается внешний вид блока *Полей выбора* и кнопок на нем.



Заголовок - заголовок блока Полей выбора; выводится в верхней части. Остальные элементы управления аналогичны элементам *Кнопки*.

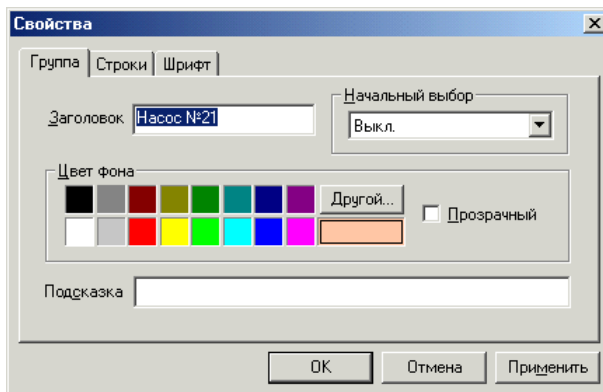
Элемент Переключатели

Элемент *Переключатели* используется для выбора одного параметра (строки) из всех параметров (строк) блока. Как и у любого элемента у блока *Кнопок-переключателей* есть события и свойства.



Элемент *Переключатели* используется для выбора одного параметра (строки) из всех параметров (строк) блока. Как и у любого элемента у блока *Кнопок-переключателей* есть события и свойства.

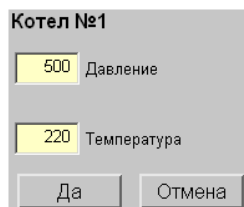
Помимо стандартных событий этот элемент имеет еще одно событие - выбор кнопки. Событие *Выбор кнопки* выполняется когда пользователь изменяет выбор переключателя. Изменение выбора переключателя производится щелчком левой клавиши мыши на требуемом параметре (строке). Ниже описаны свойства *Кнопок-переключателей выбора* в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели *Свойства*. Диалог *Свойства* контекстного меню состоит из трех закладок: *Группа*, *Строки* и *Шрифт*. На закладке *Группа* настраивается внешний вид блока *Кнопок-переключателей*.



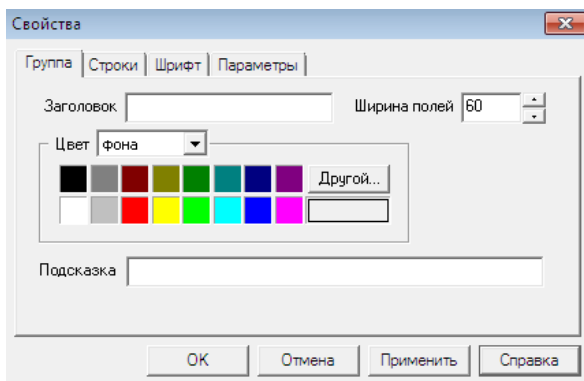
Заголовок - заголовок *Блока переключателей*; выводится в верхней части. *Начальный выбор* - строка (параметр), выбираемая по умолчанию. Остальные элементы управления аналогичны элементам *Кнопки*.

Элемент Блок параметров

Элемент *Блок параметров* представляет собой блок, объединяющий несколько полей ввода (хотя в блоке может быть одно поле ввода). Элемент *Блок параметров* используется для ввода значений параметров, входящих в данный блок. Как и у любого элемента у *Блока параметров* есть события и свойства.

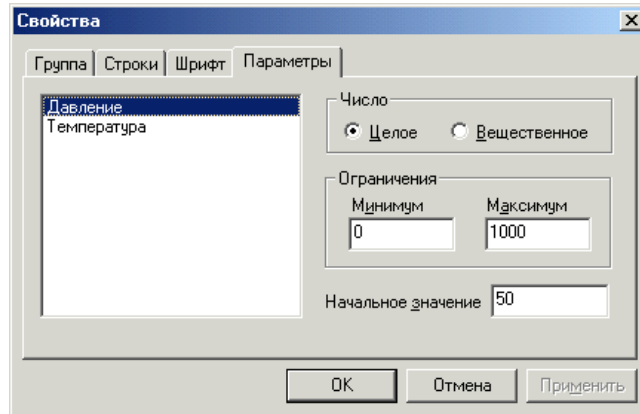


Помимо стандартных событий этот элемент имеет еще одно событие - изменение параметров. Событие *Изменение параметров* выполняется когда пользователь изменяет значение какого-нибудь параметра и нажимает кнопку *Да*. Ниже описаны свойства *Блока параметров* в диалоге *Свойства*, вызываемом из пункта контекстного меню *Свойства*. Аналогичные параметры есть на панели *Свойства*. Диалог *Свойства* контекстного меню состоит из четырех закладок: *Группа*, *Строки*, *Шрифт* и *Параметры*. На закладке *Группа* настраивается внешний вид *Блока параметров*.



Заголовок - заголовок *Блока параметров*; выводится в верхней части. *Ширина полей* - ширина полей ввода значений *Блока параметров*. Остальные элементы управления аналогичны элементам *Кнопки*.

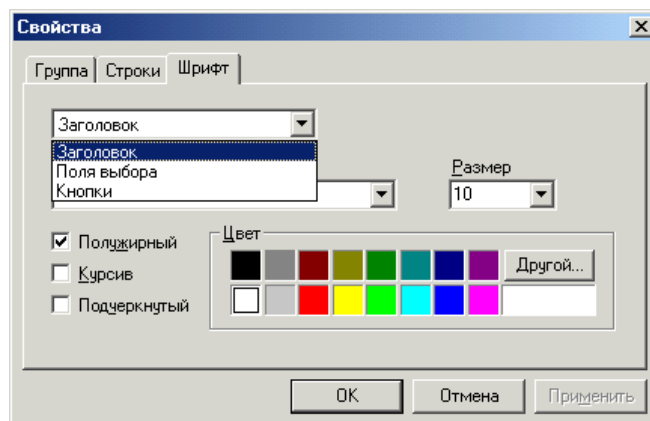
На закладке *Параметры* настраиваются свойства параметров блока.



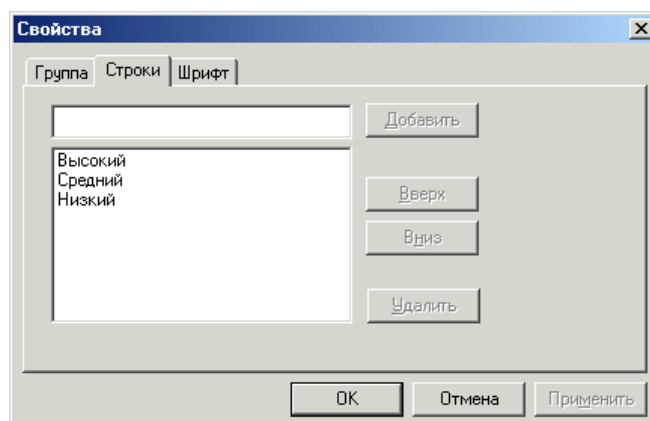
Значения параметров блока могут быть целыми либо вещественными. Для установления типа значения используйте переключатели группы *Число*. Каждое значение можно ограничить по верхней границе (максимум) и по нижней (минимум). Ограничения устанавливаются в группе *Ограничения*. Каждому параметру можно установить начальное значение (значение по умолчанию) в поле *Начальное значение*.

Общие настройки

Закладка *Шрифт* служит для определения шрифта заголовка, кнопок и строк (параметров). Список, расположенный вверху этой закладки, определяет свойство (заголовок, поля выбора, строки или кнопки), которое сейчас настраивается.



Закладка *Строки*. Для определения строк (параметров) элементов: список, выпадающий список, переключатели и поля выбора служит закладка *Строки*.



Для добавления нового параметра введите его название в верхнее поле этой закладки и нажмите *Добавить*. Кнопки *Вверх* и *Вниз* позволяют перемещать выбранный параметр на одну строку вверх или вниз списка. Для удаления параметра нажмите кнопку *Удалить*.

Элемент *Web-браузер*

Вид на мнемосхеме:



При помощи элемента *Web-браузер* в режиме исполнения можно подключиться к нужной Internet-странице. Страница свойств элемента изображена на картинке ниже.

Разное	
Url адрес	http://yandex.ru/
Исходный текст	
Источник	Url адрес
Отображать ко	Истина

Мультимедиа

Рисунок (изображение с прозрачностью) используется для создания статических изображений, загружаемых из *BMP*, *GIF* и *AVI-файлов*. Подобно другим элементам все его свойства также динамизируются. Для добавления Рисунка на мнемосхему выберете пункт *Рисунок* из группы *Мультимедиа* в *Палитре* и спроектируйте его на рабочей области экрана.

Для создания динамизации объектов мнемосхемы (мигание лампочки, повышение уровня жидкости в резервуаре и т.п.) можно создать объект, а затем динамизировать нужный параметр этого объекта. Но если необходима более сложная динамизация, то необходимо использовать *Мультфильмы* - поочередную смену изображаемых кадров. В MasterSCADA используется несколько разновидностей *Мультфильмов*:

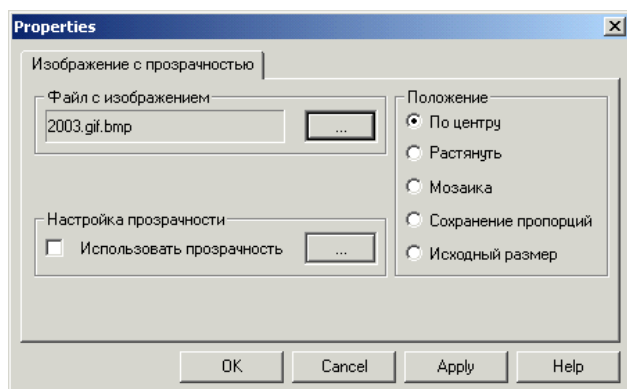
- Переключатель элементов группы;
- Проигрыватель стандартных динамических файлов в форматах *GIF* и *AVI*;
- Встроенный редактор для создания мультфильмов.

Встроенный редактор *Мультфильмов* позволяет имитировать такие эффекты, как, например, движение или вращение сложного изображения (картинки из файла), его расфокусировка, увеличение размера и т.п. Эти динамические изображения создаются на основе импортированных файлов в формате *.bmp*, *.jpg*, *.gif* и *.avi*.

Рисунки

Настройка свойств рисунка

Настройка *Рисунка* (изображения с прозрачностью) производится в диалоге *Свойства*.



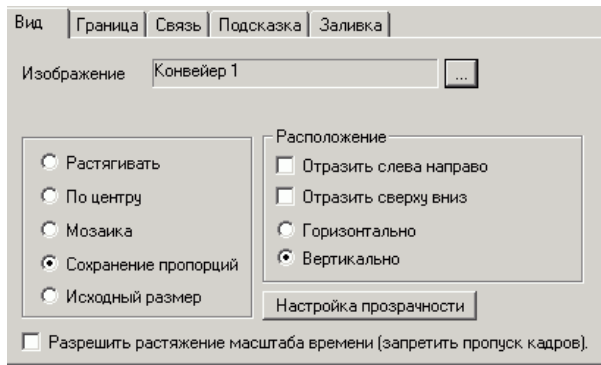
В группе *Файл с изображением* задается имя *BMP*, *GIF* или *AVI-файла*, изображение которого будет на *Рисунке*. Для выбора файла нажмите на кнопку . В появившемся стандартном диалоге открытия файла выберите требуемый файл. Данное изображение можно выводить с прозрачным фоном, т.е. этот фон не будет отображаться на мнемосхеме. Для задания прозрачности данному *Рисунку* установите флаг *Использовать прозрачность* и нажмите на кнопку .

Мультфильмы

Проигрыватель стандартных динамических файлов

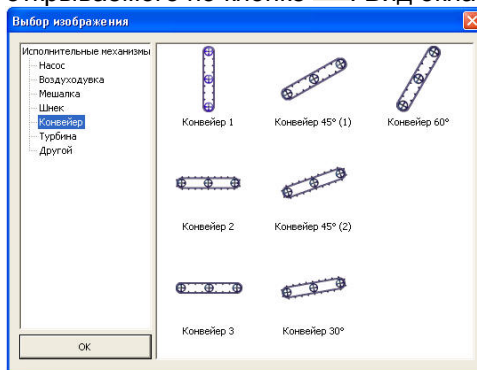
Проигрыватель стандартных динамических файлов (элемент *Мультфильм*) позволяет проигрывать в мнемосхеме файлы мультимедиа (анимированные *GIF*'ы и *AVI-файлы*). *Мультфильм* (как и другие объекты) можно создавать и настраивать 3-мя способами: визуальным проектированием в рабочей области экрана, с помощью средней части панели настройки и с помощью диалога *Свойства*. На панели элементов *Мультимедиа* или в левой части панели настройки в категории *Стандартные*, подкатегории *Мультимедиа* выберете элемент *Мультфильм*. Затем спроектируйте *Мультфильм* на рабочей области экрана. Теперь

Мультфильм расположен на мнемосхеме и имеет установленные размеры. Чтобы настроить его свойства, откройте диалог *Свойства*.



На закладке свойств *Вид* осуществляются следующие настройки:

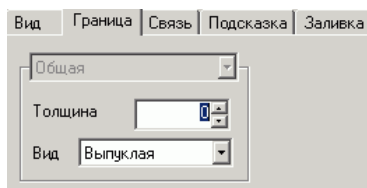
- *Изображение* - *Мультфильм* в своем составе уже имеет несколько predefined динамических изображений, выбрать которые можно в окне *Выбор изображения*, открываемого по кнопке . Вид окна показан ниже.



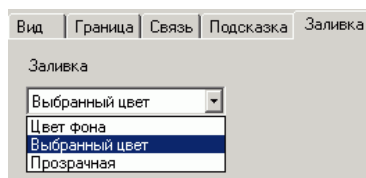
Для каждого типа изображения, перечисленного в левой части окна существует несколько вариантов, которые отображаются в правой части окна. Для выбора конкретного изображения, выделите его в правой части диалогового окна, и нажмите кнопку *Ок*. Для проигрывания другого файла нужно в окне выбрать пункт *Другой*, и с помощью появившейся кнопки *Обзор* рядом с адресом, указать нужный файл;

- *Выбор режима отображения мультфильма* - выбирается способ отображения. *Растянуть* - изображение растягивается по всей области рамки *Мультфильма*; *По центру* - изображение располагается по центру рамки с сохранением размеров оригинала; *Мозаика* - выбранное изображение тиражируется (с сохранением исходного размера) так, чтобы заполнить всю площадь, занимаемую *Мультфильмом*; *Сохранение пропорций* - изменение размеров рамки *Мультфильма* будет происходить пропорционально размерам выбранного изображения; *Исходный размер* - размер *Мультфильма* будет строго соответствовать размеру изображения;
- *Расположение*;
- *Настройка прозрачности*;
- *Установка возможности пропуска кадров* (если это необходимо).

На закладке *Граница* настраивается вид (*выпуклая*, *вогнутая*, *плоская*) и ширина границы *Мультфильма*.



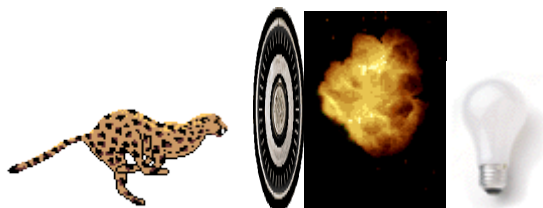
На закладке *Связь* указывается связь *Мультфильма* с объектом проекта. На закладке *Подсказка* - устанавливается текст подсказки, появляющейся в режиме исполнения, при наведении на *Мультфильм* указателя мыши. На закладке *Заливка* определяем параметры фона мультфильма.



Встроенный редактор мультфильмов

Типы мультфильмов

Показ мультфильма может производиться в прямоугольной или эллиптической области просмотра. Поэтому существуют два типа мультфильмов: мультфильмы с прямоугольной (мультфильм в прямоугольнике) и эллиптической (мультфильм в эллипсе) областью просмотра. *Прямоугольная область* просмотра может иметь различные виды рамок, вплоть до имитирующих кнопки. *Эллиптическая* - только простую рамку.



Законы динамизации изображений

С изображениями из графических файлов можно производить 5 действий: поочередная демонстрация изображений из списка, поворот на определенный угол или вращение, расфокусировка (изменение контрастности) изображения, смещение (кольцевая прокрутка) в определенном направлении (вверх, вниз, вправо и влево) и увеличение пропорционально исходным его размерам, а также проигрывать файлы мультимедиа (анимированные GIF'ы и AVI-файлы). Эти действия совершаются по определенному закону изменения изображения.

Мультфильмы (как и другие объекты) можно создавать и настраивать 3-мя способами: визуальным проектированием в рабочей области экрана и с помощью средней части панели настройки и с помощью диалога *Свойства*.

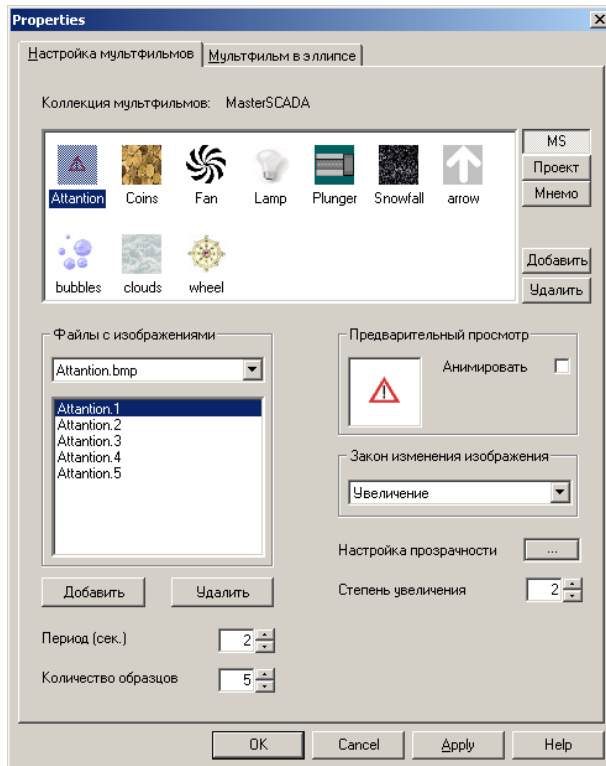
Для создания мультфильма выберите категорию *Стандартные* в левой части панели настройки и откройте в ней подкатегорию *Мультимедиа*. В этой подкатегории выберите тип мультфильма - мультфильм в прямоугольнике или мультфильм в эллипсе. Затем спроектируйте мультфильм на рабочей области экрана. Теперь мультфильм расположен на мнемосхеме и имеет установленные размеры. Чтобы добавить в него файлы растровых изображений и произвести над ними описанные выше действия (набор картинок, поворот, расфокусировку, смещение, увеличение) откройте диалог *Свойства*.

Настройка мультфильмов

В диалоге *Свойства* выберите закладку *Настройка мультфильмов*. Она присутствует и у мультфильма в прямоугольнике, и у мультфильма в эллипсе. Мультфильмы хранятся в коллекциях ресурсов. Для переключения между коллекциями в диалоге настройки существуют специальные кнопки:


- *MS* - делает текущей глобальную коллекцию;
- *Проект* - делает текущей коллекцию проекта;
- *Мнемо* - делает текущей коллекцию мнемосхемы.

При нажатии на эти кнопки, в окне отображаются мультфильмы, расположенные в выбранной коллекции. С помощью кнопок *Удалить* и *Добавить* можно редактировать состав текущей коллекции, добавляя новые мультфильмы, или удаляя имеющиеся.



В один мультфильм можно добавить несколько изображений, видео-фалов AVI или анимированных GIF'ов (хотя воспроизводится будет только одно изображение). В этом случае названия всех добавленных изображений (AVI, GIF'ов) содержатся в выпадающем списке сверху Списка файлов с изображениями, из которого можно выбрать любое другое изображение (AVI, GIF). Кнопка *Удалить* удаляет файлы изображений из данного мультфильма. Любой мультфильм проигрывается в течение заданного периода времени, а затем повторяется сначала. Период задается в поле *Период* (сек.) в секундах. Мультфильм создается либо на основе анимированных GIF'ов (AVI), либо на основе изображений по определенному закону изменения.

Существуют 5 законов изменения изображений:

- *Набор картинок* – поочередная смена одного изображения другим следующим за ним, находящемся в *Списке файлов* с изображениями. Очередность смены изображений задается кнопками  (для всех файлов, кроме анимированных GIF'ов и видео AVI). Параметром этого закона является только период времени, за который происходит смена изображений (задается в поле *Период* (сек.)). Результат динамизации можно увидеть в области предварительного просмотра, для чего установите флаг *Анимировать*. Если анимированный GIF воспроизводится с сильными искажениями, это означает, что он оптимизирован. Чтобы он воспроизводился правильно, его надо разоптимизировать. Для этой цели можно использовать практически любую из программ, используемых для создания анимированных GIF'ов;
- *Поворот изображения* - поворот выбранного изображения из списка изображений. Параметрами этого закона являются: *период времени*, за который происходит поворот (задается в поле *Период* (сек.)), *угол поворота* (задается в поле *Угол поворота* в градусах) и *количество кадров за этот период* (задается в поле *Количество образцов*). С увеличением кадров динамизация происходит плавней. При установленном флаге *Вращение* по полному кругу изображение поворачивается на 360°, т.е. описывает круг по часовой стрелке. Направление вращения против часовой стрелки задает флаг *Против часовой стрелки*. После ввода этих параметров в Списке файлов с изображениями отображаются указанное количество кадров за данный период времени. При выделении любого кадра в этом списке в области предварительного просмотра появляется его изображение. Результат динамизации можно увидеть в области предварительного просмотра, для чего установите флаг *Анимировать*;
- *Расфокусировка* – расфокусировка выбранного изображения. Параметрами этого закона являются: *период времени*, за который происходит расфокусировка (задается в поле *Период* (сек.)), *степень расфокусировки* (задается в поле *Степень расфокусировки* в процентах) и *количество кадров за этот период* (задается в поле *Количество образцов*);

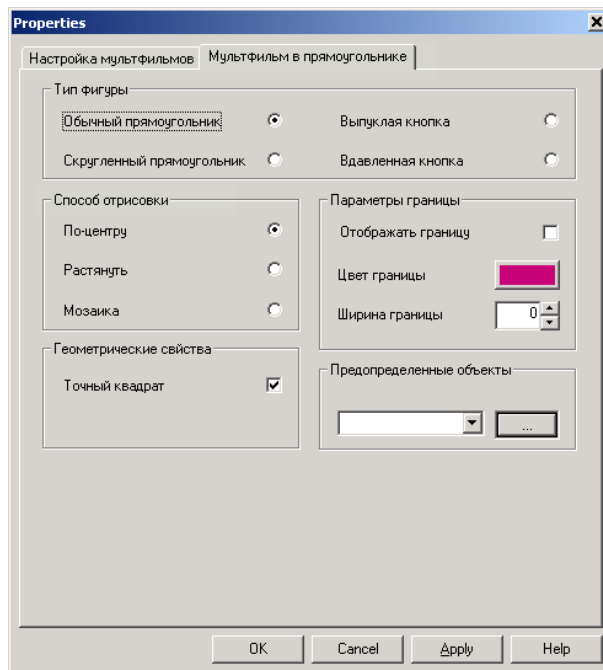
- **Смещение** – смещение выбранного изображения вправо, влево, вверх или вниз. Параметрами этого закона являются: *период времени*, за который происходит смещение (задается в поле *Период (сек.)*), *направление смещения* (задается в группе переключателей *Направление смещения* выбором направления) и *количество кадров за этот период* (задается в поле *Количество образцов*);
- **Увеличение** – увеличение выбранного изображения. Параметрами этого закона являются: *период времени*, за который происходит увеличение (задается в поле *Период (сек.)*), *степень увеличения* (задается в поле *Степень увеличения*) и *количество кадров за этот период* (задается в поле *Количество образцов*).

Для любого выбранного *Мультфильма* также можно настроить прозрачность.

Настройка области просмотра мультфильма

Данный редактор позволяет создавать мультфильмы двух типов: мультфильмы в прямоугольнике и мультфильмы в эллипсе. На самом деле прямоугольником или эллипсом является форма области просмотра мультфильма. Настроить прямоугольную область просмотра можно в средней части панели настройки или в диалоге *Свойства*. Ниже описывается диалог *Свойства*.

Выберите закладку *Настройка прямоугольника*.



Редактор позволяет создавать фактуры типа *прямоугольник* 4-х видов:


- **Обычный прямоугольник**
- **Скругленный прямоугольник**
- **Выпуклая кнопка**
- **Вдавленная кнопка**

Для изменения типа прямоугольной области просмотра мультфильма выберите тип прямоугольника в группе переключателей *Тип фигуры* и нажмите *Ок*.

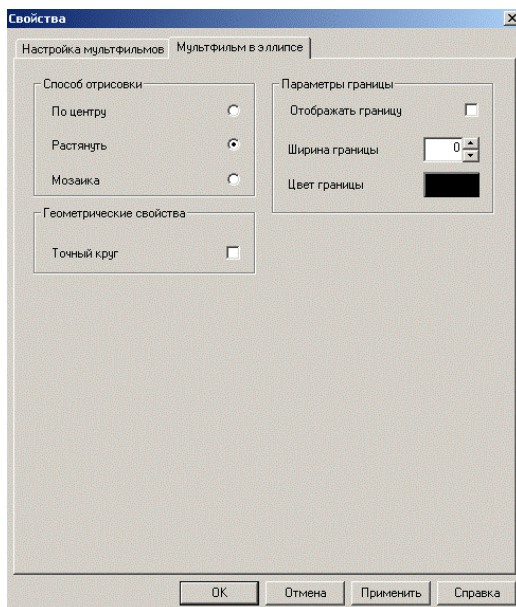
В группе *Параметры границы* задаются цвет и ширина прямоугольника области просмотра, а также его видимость на экране. При создании нового мультфильма этот флаг сброшен, т.е. граница не отображается. Для отображения границы установите флаг и нажмите кнопку *Ок*.

В группе *Геометрические свойства* можно задать дополнительные настройки:

- Мозаичное заполнение области просмотра данным образом (необходимо, если размеры изображения не превышают размера области просмотра);
- Изменение прямоугольного размера области просмотра на квадратную;
- Определение радиуса скругления (для скругленного прямоугольника).

Настройки прямоугольника можно сохранять в файлах коллекций объектов, а затем использовать при создании новых объектов. Сохранение и загрузка файлов коллекций осуществляется аналогично предопределенным материалам и производится из диалога *Фигуры*, который открывается при нажатии на кнопку .

Свойства эллиптической области просмотра похожи на свойства прямоугольной. Откройте диалог *Свойства* и выберите в ней закладку *Настройка эллипса*.



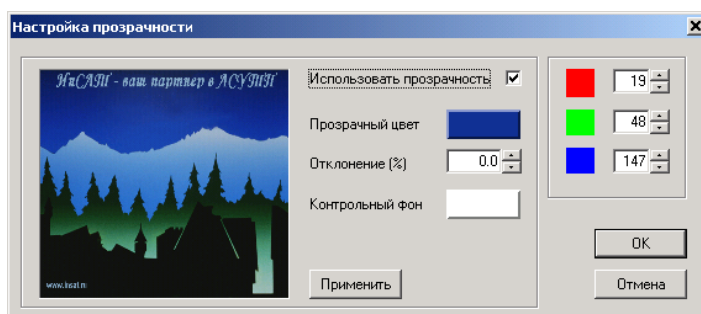
На этой закладке задаются геометрические свойства и параметры границы *Эллипса*:

- Мозаичное заполнение области просмотра данным образом (это имеет смысл, если размеры изображения не превышают размера области просмотра);
- Изменение эллиптического размера области просмотра на круглый.

Настройка параметров границы эллипса производится аналогично прямоугольнику.

Настройка прозрачности

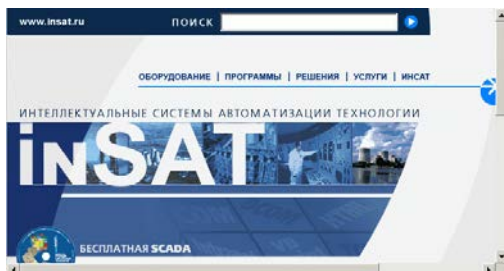
Как у мультфильма, так и у рисунка можно задать фон прозрачности, т.е. тот цвет изображения, который при наложении его на другие объекта мнемосхемы не отображается (становится *Прозрачным*). Для определения прозрачности образца нажмите на кнопку *Настройка прозрачности* на странице свойств мультфильма/рисунка. Откроется диалог *настройка прозрачности*.



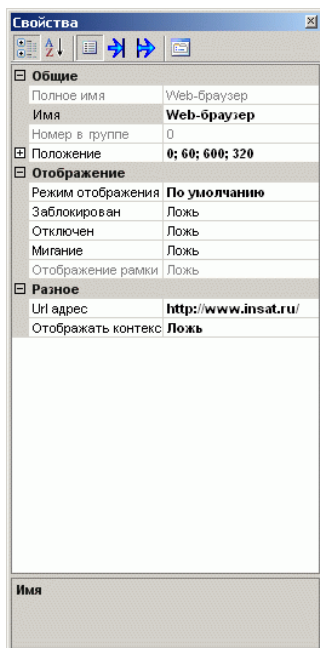
Чтобы выбрать прозрачный фон достаточно щелкнуть на этом цвете в окне просмотра изображения и установить флаг *Использовать прозрачность*. Нажав кнопку *Применить*, мы видим как прозрачность будет выглядеть на экране, для чего прозрачный фон закрашивается другим цветом (контрольный фон). Если результат неподходящий (наличие очерчивающей рамки объекта, "снег" в изображении), задайте процент отклонения от данного цвета в поле *Отклонение (%)*. Завершите настройку прозрачного фона нажатием на кнопку *Ок*.

Web-браузер

Элемент предназначен для отображения *Интернет страниц* на мнемосхеме MasterSCADA.



Для настройки элемента выделите его на мнемосхеме и перейдите на панель *Свойства*.



Для этого элемента характерны следующие настройки:

- *Url адрес* - задается адрес страницы в сети Интернет;
- *Отображать контекстное меню* - если установлено значение *Истина*, то тогда в режиме исполнения, при нажатии правой кнопки мыши, будет отображаться стандартное контекстное меню Интернет страницы, позволяющее сохранять, добавлять в избранное, и т.п.

Эти свойства можно динамизировать по входу.

Метрологическая поверка

Модуль *Метрологическая поверка* для MasterSCADA ставит перед собой следующие цели:

- Упростить процесс поверки переменных (записи значений);
- Автоматически создать и напечатать документы для метрологической аттестации.

Работа осуществляется в три этапа описанные ниже.

Настройка методик поверки. Каждая методика поверки соответствует группе измерительных каналов (ИК) и связана со шкалой MasterSCADA. Связь методики и шкалы предоставляет следующие преимущества:

- Часть настроек шкалы (например, минимум и максимум) автоматически становятся настройками группы ИК, чем снимается необходимость в их повторном установлении;
- Переменные, которым назначена шкала, автоматически становятся принадлежащими определенной группе ИК. Таким образом, без дополнительных действий, системе известно о количестве ИК в каждой группе.

В настройку методики входит:

- Задание контрольных точек;
- Задание свойств методики, необходимых для проведения статистических вычислений, или для заполнения определенных полей в аттестационных документах.

Поверка переменных. При поверке записываются значения переменных, соответствующие контрольным точкам методики. Записанные величины помещаются в отдельный архив, из которого считываются впоследствии для выполнения статистических расчетов и помещения в аттестационные документы. Очевидно, что автоматизировать полностью этот процесс не представляется возможным, но с модулем *метрологическая поверка* делать все становится намного удобнее, чем вручную. У каждой поверяемой переменной есть специальная закладка, на которой отображается список контрольных точек, таблица значений и другая полезная функциональность. Текущие значения переменной записываются одним нажатием кнопки.

Формирование отчетов. Для метрологической аттестации необходимо предоставить определенный пакет документов. Некоторые из них индивидуальны для каждой системы управления и могут быть составлены только с участием разработчиков системы на месте проведения аттестации. Но большинство документов, имеет сходную для всех систем структуру, такие отчеты автоматически формируются с помощью MasterSCADA. В них помещаются сведения о системе, поверяемых переменных, результаты статистических вычислений.

MasterSCADA формирует отчеты на основе заранее созданных шаблонов, хранящихся на компьютере в известном месте и имеющих формат, доступный для редактирования. Возможно, если необходимо, корректировать шаблоны, оставляя неизменным содержимое ссылок полей, предназначенных для автоматического заполнения. Таким образом, наличие опции метрологической поверки существенным образом сокращает время и усилия, затрачиваемые на подготовку системы АСУТП к метрологической аттестации.

Порядок действий при настройке и проведении метрологической поверки

Создание шкал. В дереве *Системы* выделите элемент *Система* - закладка *Шкалы* - лист *Аналоговые* - таблица *Аналоговые значения*. Создайте и настройте *Интервалы* для всех переменных, которые необходимо поверить. Предполагается, что все переменные, имеющие шкалу с настроенной методикой поверки должны быть поверены, это необходимо учитывать при создании шкал. Если в системе уже есть подходящий интервал, но его планируется использовать для переменных, которые не надо поверять, создайте еще один такой же интервал под другим именем специально для поверяемых переменных.

Настройка отключенных приборов. Во время проведения поверки некоторые устройства отключаются и вместо них помещают эталонные приборы. Тем не менее, погрешность этих устройств необходимо учитывать. Нужно вписать названия этих приборов и их паспортные

погрешности, чтобы эти данные вошли в отчеты. Для удобства устройства группируются в списки. В дереве *Системы* выделите элемент *Система* - закладка *Шкалы* - лист *Метрология* - группа *Списки устройств, не участвующих в эксперименте*. Создайте необходимое количество наборов устройств, которые будут отключены, задайте им имена. Для каждого набора создайте список устройств и погрешностей.

Настройка методик поверки. Все переменные, имеющие один и тот же интервал поверяются одинаковым образом, то есть имеют одну и ту же *Методику поверки*.

Понятие методика поверки включает в себя:

- Набор опорных точек;
- Число измерений;
- Погрешность измерительного канала;
- Список устройств, не участвующих в эксперименте (отключенных во время поверки, см. п. настройка отключённых приборов).

В списке поверяемых шкал последовательно настройте свойства методики поверки для каждого интервала, переменные которого нужно поверить. Интервалы, для которых нет методики поверяться не будут. Для этого выделите в списке интервал, нажмите кнопку *Создать/изменить методику*. В открывшемся окне создайте список опорных точек и отредактируйте остальные свойства.

Выбор шкал у поверяемых переменных. Для каждой переменной, которую нужно поверять проделайте следующее:

- Перейдите по следующему пути: в дереве объектов выделите *переменную* - закладка *Общие*;
- Выберите Диапазон измерения;
- Всем переменным, которые надо поверить нужно выбрать соответствующий *Диапазон измерения*. После этого переменной автоматически будет назначена методика, настроенная для выбранного диапазона (интервала).

Дополнительное редактирование отчетов. Просмотрите шаблоны отчетов и, если необходимо, внесите какие-то изменения.

Запись значений. В режиме исполнения для каждой переменной, которую нужно поверять проделайте следующее:

- Перейдите по следующему пути: в дереве объектов выделите *переменную* - закладка *Поверка* (в режиме разработки этой закладки нет);
- На этой закладке будет список опорных точек (настроенный в п. настройка отключённых приборов), для каждой опорной точки проделайте следующее:
- выделите опорную точку в списке *Выбор контрольной точки*;
- установите с помощью образцового прибора нужное образцовое значение;
- запишите необходимое количество измерений (мгновенных значений переменной).

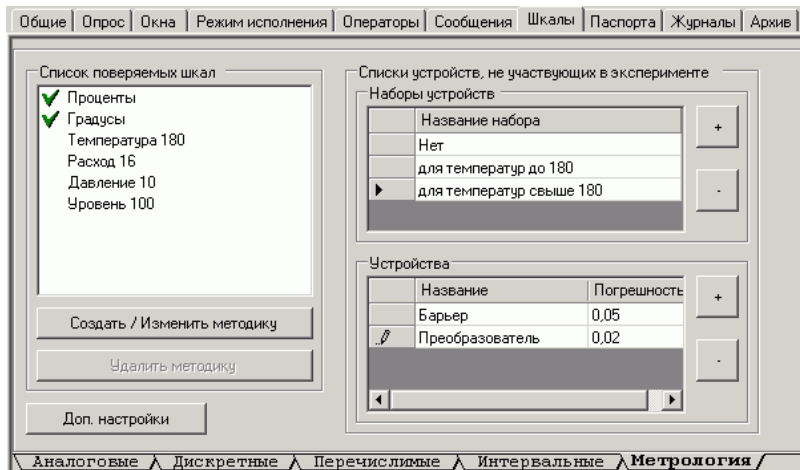
Формирование отчетов. В режиме исполнения перейдите по следующему пути: в дереве системы выделите элемент *Система* - закладка *Шкалы* - лист *Метрология* (в режиме разработки он выглядит по-другому). Выберите нужный элемент для отчета, выберите вид отчета, нажмите кнопку *Сформировать*.

Общие сведения

Операции с методиками поверки

Чтобы сделать переменную MasterSCADA *поверяемой*, необходимо выполнение всего двух условий: чтобы у переменной была выбрана шкала и у этой шкалы должна быть методика поверки. Чтобы создать и настроить методику выполните действия указанные ниже.

Выделите в дереве системы MasterSCADA в режиме разработки корневой элемент *Система*, у *Системы* откройте закладку *Шкалы*. Если шкалы для переменных проекта ещё не созданы, необходимо сделать это, по крайней мере для аналоговых шкал, так как методика поверки - это свойство шкалы и, не имея шкалы, нельзя настроить методику. Если шкалы созданы, можно открыть лист *Метрология*.



В первую очередь необходимо настроить списки устройств, не участвующих в эксперименте. Это те устройства, через которые не проходит сигнал калибратора, но погрешность которых необходимо учитывать. Для удобства такие устройства группируются в списки, и при настройке методики поверки можно будет выбрать сразу список, содержащий несколько устройств. Слева на листе *Метрология* можно видеть список всех аналоговых шкал проекта. В этом списке выберите шкалу, переменные которой в будущем необходимо будет поверять и нажмите кнопку *Создать*. После этого откроется окно настроек методики. Установив в окне все необходимые свойства, нажмите *ОК* - окно будет закрыто и напротив шкалы появится значок, говорящий о том, что шкала имеет методику. Теперь все переменные проекта, для которых выбрана эта шкала автоматически подлежат поверке.

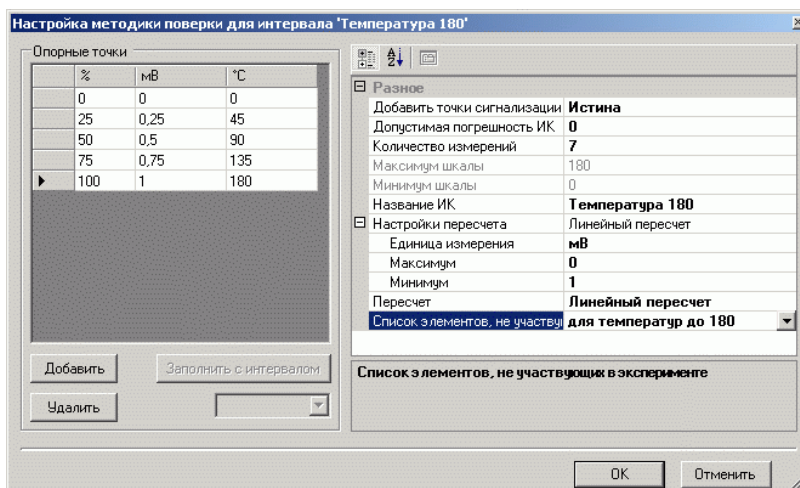
- ❖ **Важно!** При составлении свидетельства об аттестации, в нем указывается общее число переменных, подлежащих поверке и какое количество из них действительно было поверено, это количество должно соответствовать минимальному объему выборки. Если в проекте какие-либо переменные используют данную шкалу, но поверять их не предполагается, просто создайте две одинаковых шкалы, а методику настройте только для одной из них.

Если методика была создана для шкалы ошибочно, ее можно удалить. Настройки методики всегда можно изменить, нажав кнопку *Изменить*. После того, как для шкалы создана методика поверки, у переменных в режиме исполнения появится закладка *Поверка*, если им назначена эта шкала.

Настройка методики поверки

Настройка методики включает в себя:

- Установление значений свойств методики;
- Создание списка контрольных точек методики.



Свойства методики. Каждая методика поверки обладает определенными свойствами, например *Имя группы измерительных каналов*. Список этих свойств отображается в правой части окна редактирования методики в специальной таблице. Для изменения какого-либо

свойства достаточно выделить его и ввести новое значение. При этом в нижней части таблицы отображается имя, и возможно, дополнительная информация о свойстве.

Контрольные точки. Контрольные (или опорные) точки - это ожидаемые значения переменной, при которых производятся контрольные измерения. Они вводятся пользователем в процентах от шкалы, например 0%, 25%, 50%, 75%, 100%.

Таблица для ввода контрольных точек содержит три столбца:

- Значения в процентах, которые следует ввести;
- Значения образцового прибора (например, термодары). Эти значения могут быть рассчитаны автоматически, в соответствии со способом пересчета, который выбирается в свойствах методики. Если же в свойствах выбран пересчет *вручную*, потребуются ввести значения в таблицу самостоятельно. Этим можно воспользоваться в случае нестандартной системы пересчета. Значения образцового прибора необходимы для составления отчетных документов;
- Значения переменной, которые заполняются автоматически, на основе %, линейно в соответствии с минимумом и максимумом шкалы. Эти значения и являются ожидаемыми при поверке. На основе разности реальных значений и ожидаемых рассчитывается погрешность каналов.

Поверка переменной

При поверке переменной нужно произвести указанное количество измерений для каждой контрольной точки. Значения измерений записываются и помещаются в архив поверки, на основе которого составляются отчетные документы. Чтобы произвести поверку переменной в режиме исполнения выделите ее в дереве MasterSCADA и откройте закладку *Поверка*.

%	%	Среднее
0	0	2,42857
10	10	11,8571
50	50	

№	Значение
1	42
2	48
3	49
4	44
5	
6	
7	

В левой части закладки находится таблица контрольных точек, содержащая три столбца:

- Значение контрольной точки в %;
- Ожидаемое значение переменной в единицах шкалы;
- Среднее значение измерений. Этот столбец заполняется автоматически после того, как для контрольной точки проведено нужное число измерений (то есть записано нужное число значений). Если столбец не заполнен, значит измерений нет или их недостаточно.

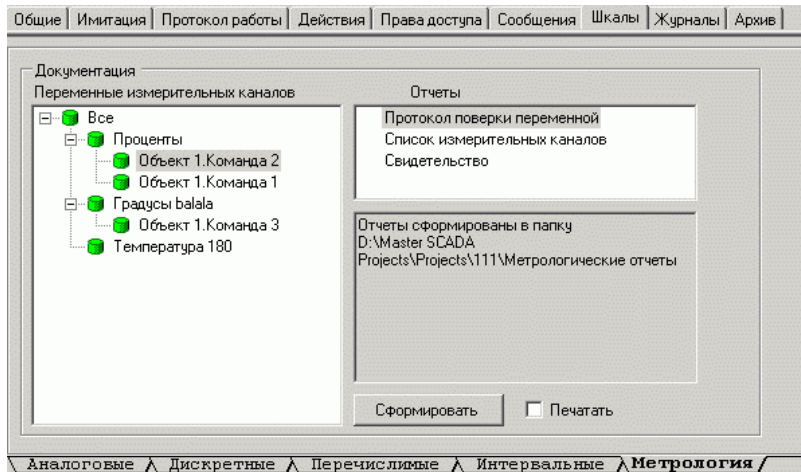
В таблице контрольных точек следует выделять точки и производить записи значений. Запись значений для каждой точки делается в правой части страницы. Для этого достаточно просто нажать кнопку *Записать* нужное число раз (или нажать кнопку *Записывать автоматически*). Если была допущена ошибка при записи значений (например, выяснилось, что образцовый прибор не был настроен), нажмите *Сбросить точку* - значения для данной точки будут удалены. Чтобы удалить записи для всех точек переменной (например, если переменная была поверена, а после этого пришлось изменить условия поверки или свойства шкалы) - нажмите кнопку *Сбросить все*.

Записанные значения переменной обновляются в архиве только в случае, если проведены все измерения всех контрольных точек. При этом кнопка *Записать* внизу закладки доступна для нажатия. Не забудьте, что если кнопка доступна, и осуществляется переход на другую

страницу настроек, или выделяете в дереве другой элемент, то она будет нажата автоматически.

Формирование отчетов

После поверки переменных в необходимом объеме, можно приступить к завершающему этапу - созданию документов для метрологической аттестации. Для этого выделите в дереве системы MasterSCADA в режиме исполнения корневой элемент *Система*, у *Системы* откройте закладку *Шкалы*, на закладке откройте лист *Метрология*.



Слева на странице находится дерево, в котором все переменные проекта, подлежащие поверке сгруппированы по шкалам (каждая шкала соответствует группе измерительных каналов). Справа находится список отчетов, которые можно сформировать автоматически. Для создания отчета произведите следующие действия:

- Выделите в левой части элемент, для которого создается отчет (переменную, группу каналов или все);
- В правой части выделите отчет, который хотите создать;
- Если нужно не только сохранить документы, но сразу же отправить их на печать, поставьте галочку *Печать*;
- Нажмите кнопку *Сформировать*.

Если нет никаких ошибок, нужный отчет, или группа отчетов будут сформированы и сохранены в определенную папку на компьютере. Всплывёт сообщение о том, что отчеты успешно сформированы и путь к этой папке (ее расположение можно менять в дополнительных настройках). При выделении элемента и отчета, необходимо знать следующее: если отчет предназначен для вложенного элемента, а выделен родительский, то будут сформированы документы для каждого из вложенных элементов. Например, если слева выделить корневой элемент, а отчет выбрать для переменной (*протокол поверки*), то один за другим автоматически будут формироваться документы для всех переменных проекта.

Дополнительные сведения

Поверка системы из нескольких компьютеров

Если в системе используются несколько компьютеров, поверка переменных и формирование отчетов осуществляется обычным порядком, но выполнять все действия необходимо только на одном компьютере. Если переменная принадлежит объекту, относящемуся к другому компьютеру, с ней автоматически будет установлена связь и ее значения должны записываться на данном компьютере в файл архива метрологической поверки.

Архив записанных значений хранится в одном файле. При необходимости его можно найти *Папка_проекта\Система\Metrology\metrologyarchive.xml*, но редактировать этот файл крайне нежелательно. Значения, используемые при формировании отчетов берутся из архива, поэтому и должен использоваться один компьютер. Таким образом, если у имеется система из нескольких компьютеров, выберите с самого начала один из них, на нем произведите поверку всех переменных, и с него же необходимо формировать документы.

Шаблоны в формате Excel

Шаблоны, используемые при подготовке отчетов расположены на компьютере по примерно следующему пути: `...\Program_Files\InSAT\MasterSCADA\Config\Metrology`. Шаблоны, содержат типичное расположение данных в документах для метрологической аттестации, но их можно изменить. При этом следует обратить внимание на следующие детали:

- Шаблоны устанавливаются вместе с MasterSCADA и хранятся в папке с установленной программой. Это значит, что внесенные изменения, необратимо отразятся на всех проектах, поэтому рекомендуется на всякий случай перед редактированием скопировать их в другую папку, чтобы при необходимости можно было восстановить исходный вариант;
- Некоторые ячейки документов содержат ссылки на MasterSCADA. Если не запускать программу и не обновлять их, они будут обозначены *REF*. Ячейки возможно передвигать и копировать, но если изменить текст ссылки, она может стать недоступной;
- Некоторые поля содержат ссылки на таблицы из внешних файлов. С ними надо быть особенно внимательными, так как они могут быть никак не обозначенными.

Список шаблонов и их описание содержится в файле *templates.xml* (в той же директории где шаблоны). Каждый шаблон описан там примерно в таком виде: *Template name="Свидетельство" file="attestationcertificate.xls" filter="System"/*, где атрибуты обозначают следующее:

- *name* - название документа для пользовательского интерфейса;
- *file* - название файла шаблона;
- *filter* - объект отчета, может принимать значения:
 - *System* - один документ для всего проекта;
 - *Variable* - по одному документу для каждой переменной;
 - *MeasuringChannel* - по одному документу для каждой группы переменных.
- ❖ **Важно!** Возможно создать простой собственный шаблон, используя имеющиеся в готовых шаблонах ссылки, и если сохранить его в указанной папке и добавить его описание в *xml*-файл, он будет доступен для формирования и печати из MasterSCADA, однако необходимость в этом возникает редко.

Внутренний механизм формирования отчетов

В данной статье описывается что на самом деле происходит за кадром при автоматическом формировании отчетов. Эта информация может оказаться полезной, и в ряде случаев помочь при разрешении конфликтов в системе. Когда пользователь нажимает кнопку *Сформировать* автоматически должны быть выполнены следующие действия:

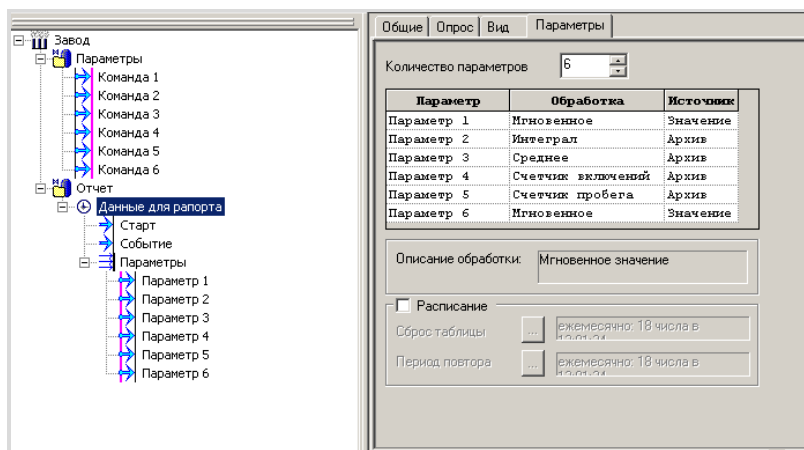
- Определить, нет ли уже открытых приложений *MS Excel*, и вывести предупреждение (эта операция успешно заканчивается не во всех операционных системах. В любом случае лучше закрыть *Excel* перед автоматическим формированием отчетов);
- Определить объект отчета (один или несколько), заданный в файле описания шаблонов атрибутом *filter*. Объектом отчета может быть переменная (или несколько), группа *ИК* (или несколько), или проект в целом;
- Далее для каждого объекта отчета:
 - Файл шаблона скопировать во временную папку (*Папка_проекта\Система\Temp*)
 - Создать во временной папке все текстовые таблицы, относящиеся к объекту отчета;
 - Открыть копию шаблона во временной папке. При открытии *Excel* запрашивает у MasterSCADA значения ссылок на нее и обновляет содержание полей, ссылающихся на текстовые таблицы. *Excel* открывает файл в невидимом режиме;
- Дать команду программе *Excel* *Сохранить как Web-страницу* файл в директорию, которая указывается в дополнительных настройках метрологической поверки. Имя файла составляется из названия объекта отчета и названия шаблона;
- Если стоит галочка *Печать*, дать команду *Печать* программе *Excel*;
- Дать команду *Закрыть* (файл) программе *Excel*;
- Завершить работу *Excel*;
- Вывести в текстовое поле сообщение о том, что отчеты успешно сформировались.

Если во время выполнения этих действий вдруг происходят какие-то ошибки, информация об этом также выводится в текстовое поле.

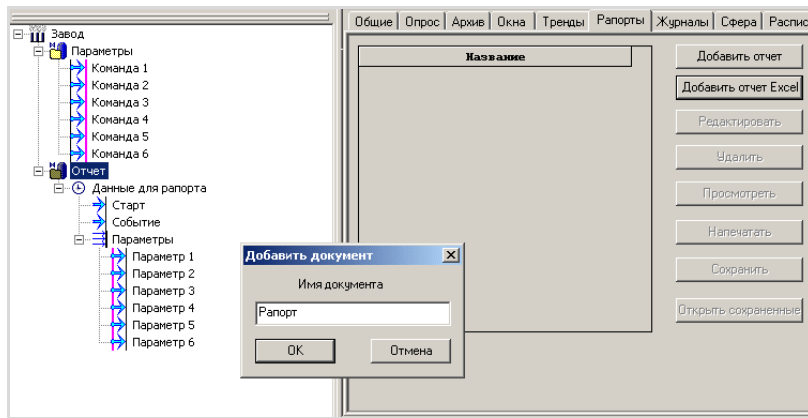
Рекомендации по созданию отчетов

Создание рапортов с использованием ФБ *Данные для рапорта* из категории *Работа с архивом* происходит по алгоритму описанному ниже. ФБ *Данные для рапорта* работает с данными, которые поступают в настоящий момент времени.

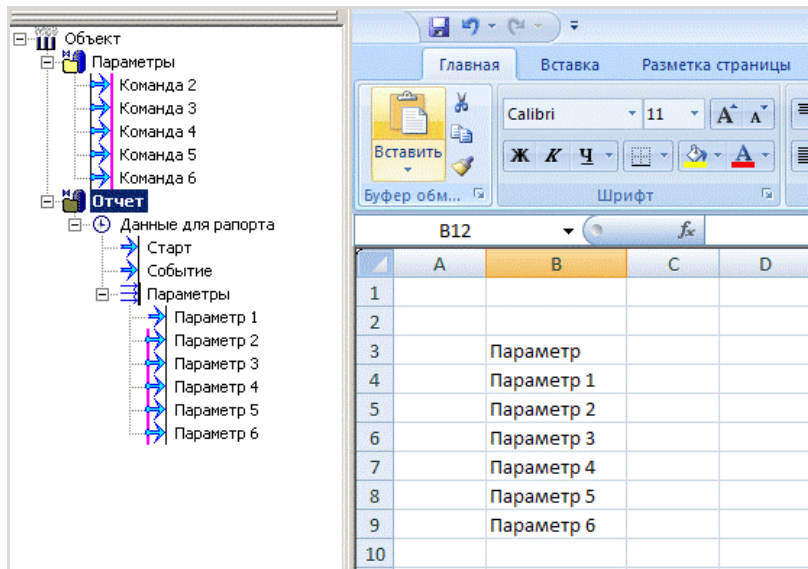
- Создадим *Объект*, который назовем *Параметры*;
- В данный объект добавим три аналоговые команды и три дискретные. Для всех команд проставить имитацию;
- В корневом объекте создадим еще объект, который назовем *Отчет*;
- Добавляем в дерево объектов ФБ *Данные для рапорта* из категории *Работа с архивом*;
- На странице свойств *Параметры* функционально блока определяем количество параметров (в нашем случае 6 параметров);
- Связываем шесть команд и шесть параметров;
- На странице свойств *Параметры* настраиваем тип обработки параметров:
 - *Обработка* - позволяет выбрать тип обработки данных для расчета значения параметра. В случае, если в качестве источника данных используется архив (задается в столбце *Источник*), для расчета значения параметра можно использовать архивные функции (для аналогового входного значения – среднее, интеграл; для дискретного – счетчик включений(сколько раз переменная перешла из состояния ноль в состояние единица), счетчик пробега(сколько времени переменная находилась в состоянии единица)), функциональность которых аналогична соответствующим ФБ;
 - *Источник* - выбор источника данных для расчета значения параметра. Если выбрано *Значение*, то используется мгновенное значение соответствующего входа. Если выбрано *Архив*, то значение параметра рассчитывается в соответствии с типом обработки (задается в столбце *Обработка*);
- В поле *Расписание* установите период, с которым производится запись (например, один раз в час - для сменного рапорта) и период формирования полного рапорта (например, раз в смену);



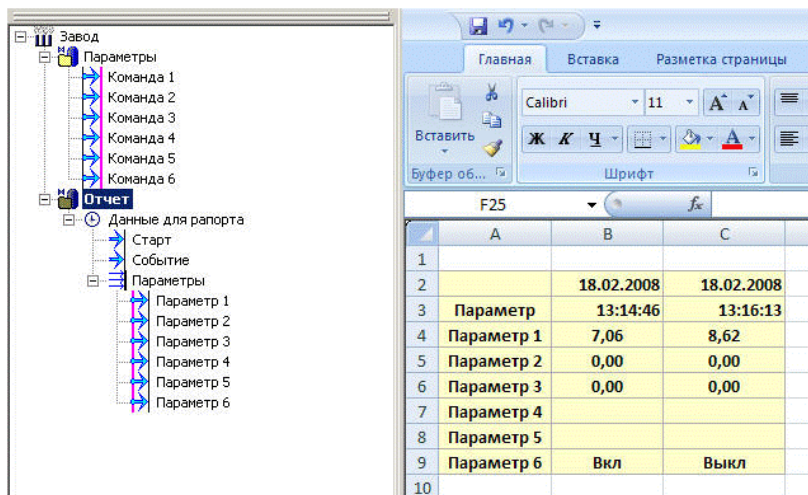
- Выделите объект, для которого создается рапорт;
- Откройте закладку *Рапорты*;
- Нажмите кнопку *Добавить*;
- Введите имя рапорта и нажмите *ОК*.



- Перетащите ФБ *Данные для рапорта* в *Рапорт* (таблица *Excel*).

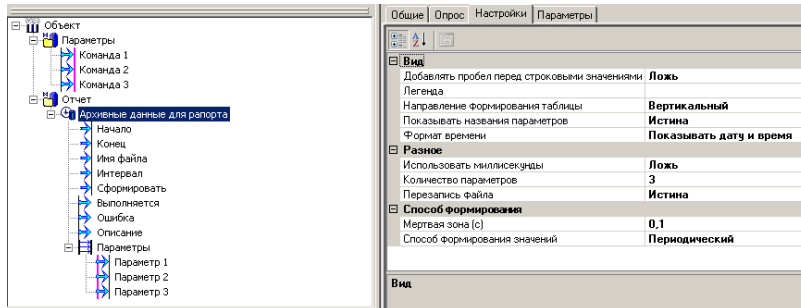


Все формулы и диаграммы *Excel* вычисляются по текущим данным. Для полей с датой и временем необходимо задать соответствующий формат ячеек. В режиме исполнения готовый рапорт может печататься или отображаться по событию или расписанию. На странице свойств *Вид* можно задать вид рапорта. Переменные *Старт* и *Событие* могут заменить расписание для ФБ *Данные для рапорта*. Когда приходит передний фронт на переменную *Событие*, в отчет пишется либо новая строка, либо новый столбец, в зависимости от настройки рапорта. Когда приходит передний фронт на *Старт* все данные обнуляются. В режиме исполнения подаем на вход *Событие* несколько импульсов - в отчет записываются данные.

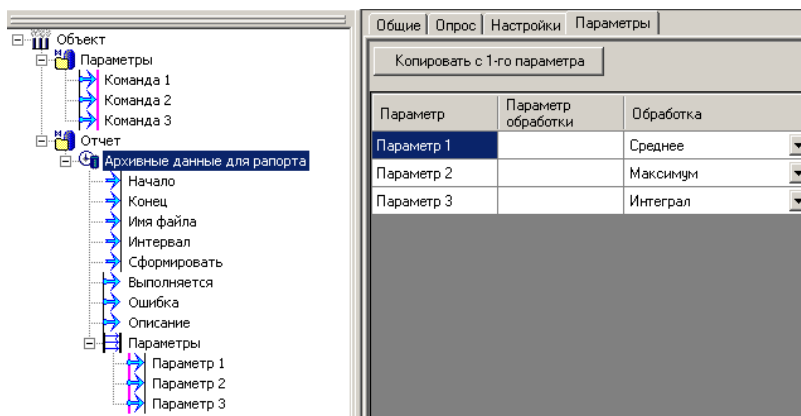


Создание рапортов с использованием ФБ *Архивные данные для рапорта* из категории *Работа с архивом* происходит по алгоритму описанному ниже. ФБ *Архивные данные для рапорта* работают только с архивными данными.

- Вставьте из палитры в дерево *Объектов* функциональный блок *Архивные данные для рапорта*;
- На странице свойств *Настройки* функционального блока определите количество параметров, и способ формирования значений: периодический, по изменению, по событию, итоговый;
- Свяжите параметры с командами;
- Для каждой команды проставить галочку *Архивировать*;



- На странице свойств *Параметры* укажите способ обработки каждого параметра: последнее значение, интеграл, среднее, интегральное среднее, максимум, минимум, интерполяция;
- При выборе обработки *Интеграл* или *Интегральное среднее*, задается параметр обработки (в секундах).



Описание *Входов*:

- *Начало* - нижняя граница интервала времени, за который берутся архивные значения переменных;
- *Конец* - верхняя граница интервала времени, за который берутся архивные значения переменных;
- *Имя файла* – путь (относительный или абсолютный) к файлу, где будет храниться отчет. Файл должен иметь расширение .csv. Если путь не указан, то файл по умолчанию формируется в папку проекта, в подпапку, соответствующую этому ФБ. Если задан относительный путь, то он применяется к папке по умолчанию. В режиме разработки можно задать путь в поле *Константа* на закладке *Опрос*;
- *Интервал* - с данным периодом будут выбираться интервалы времени, для которых будет производиться статистическая обработка и вывод результатов в рапорт. Значение входа задается в секундах и должно быть больше нуля;
- *Сформировать* - по переднему фронту производится заполнение рапорта в соответствии со значениями входов ФБ.

Описание *Выходов*:

- *Выполняется* - принимает значение *Истина* во время формирования отчета;
- *Ошибка* - если рапорт не удалось сформировать, то выход равен *Истина*;
- *Описание* - выдается текст ошибки.

Группа *Параметры* содержит входы параметров, которые формируют рапорт. Переменные, связанные с этими входами должны архивироваться.

- ❖ **Важно!** Если способ формирования отчета не периодический, а по событию, то появляется вход *Событие*. К данному входу должна быть подключена дискретная

архивируемая переменная. Каждый раз, когда переменная переходит из 0 в 1, в рапорте формируется новая строка.

Для ФБ *Архивные данные для рапорта* - рапорт создается точно так же, как и для ФБ *Данные для рапорта* (в объекте на закладке *Рапорты*). ФБ *Архивные данные для рапорта* может так же, как и ФБ *Данные для рапорта*, перетаскиваться в *Excel*.

Служба шаблонных объектов

Служба шаблонных объектов отвечает за создание связи между объектом шаблоном и объектом экземпляром и последующую автоматизированную синхронизацию экземпляров на основе шаблона. Объект шаблон является эталонным объектом одного или более объектов экземпляров. Для назначения объекта шаблоном нужно установить флаг *Является шаблоном* на закладке *Общие* у объекта, после этого у объекта появляется закладка *Шаблон*.

Для создания экземпляра возможно:

- Дублировать шаблонный объект;
- Копировать шаблонный объект в буфер с последующей вставкой из буфера;
- Перетащить шаблонный объект в поле *Шаблон* на закладке *Общие* у объекта.

После назначения объекта экземпляром у него появляется страница свойств *Экземпляр*. Для разрыва связи между экземпляром и шаблоном нужно нажать на кнопку *X* справа от поля *Шаблон* на закладке *Общие* у объекта. После создания связей шаблон – экземпляры, различия между шаблоном и экземплярами можно просматривать на закладках *Шаблон* и *Экземпляр*. На каждой из этих закладок отображается дерево различий.

Для внесения изменений нужно выбрать те изменения, которые необходимо синхронизировать, и нажать одну из трех кнопок:

- *Изменить экземпляры* - данная кнопка находится на странице *Шаблон* и вносит выбранные изменения во все экземпляры данного шаблона;
- *Изменить экземпляр* - данная кнопка находится на странице *Экземпляр* и вносит выбранные изменения в текущий экземпляр данного шаблона;
- *Изменить шаблон* - данная кнопка находится на странице *Экземпляр* и вносит выбранные изменения из текущего экземпляра в шаблон.

Возможно построение сложной иерархии шаблонов и экземпляров. Один объект может являться как шаблоном, так и экземпляром.



Адрес: 123298, г. Москва, ул. Маршала Бирюзова, д. 1, корп. 3

Телефон: +7 (495) 989-22-49

Факс: +7 (499) 943-02-14

Email:

Отдел продаж программного обеспечения - scada@insat.ru

Служба технической поддержки пользователей - support@insat.ru

Отдел продаж оборудования – sales@insat.ru

Наши сайты:

www.insat.ru

www.masterscada.ru

www.masteropc.ru

www.sky-monitor.ru

www.sky-vent.ru