

## Создание системы сообщений

### 6.1 Основные сведения

#### 6.1.1 Вывод системных сообщений и сообщений процесса

##### Введение

- **Сообщения, определяемые пользователем**  
Сообщения процесса определяются пользователем и предназначены для информирования о состоянии процесса или об измеренных данных процесса, которые передаются с ПЛК на устройство HMI.
- **Системные сообщения**  
Системные сообщения предназначены для информирования об особых состояниях устройства HMI или ПЛК и представляют собой готовый набор стандартных сообщений для каждого из устройств.

Появление и пользовательских, и системных сообщений инициируется устройством HMI или ПЛК. Сообщения могут выводиться на устройство HMI.

##### Задачи системы сообщений

- **Вывод сообщений на устройстве HMI:** Для информирования о состояниях процесса и событиях, возникающих в технологическом процессе или на установке.  
Сообщения появляется сразу же после перехода технологического процесса в некоторое состояние.

## 6.1 Основные сведения

- Формирование отчетов: Сообщения и информация о событиях сообщений выводится на печать.
- Запись сообщений в журналы (архивирование сообщений): Данные сообщений сохраняются в журналах сообщений с целью последующего их редактирования или анализа.

### 6.1.2 Сообщения, определяемые пользователем

#### 6.1.2.1 Возможные процедуры сообщений

##### Вывод сообщений в WinCC flexible

Процедура для вывода сообщения, называемая также процедурой сообщения, определяет тип информации, которая инициирует вывод сообщения и тем самым задает свойства сообщения.

WinCC flexible предоставляет следующие возможности инициации сообщений:

- Процедура дискретного сообщения  
Сообщение выводится на устройство HMI, если в ПЛК установлен определенный бит. С этой целью в WinCC flexible создаются дискретные сообщения
- Процедура аналогового сообщения  
Сообщение выводится на устройство HMI, если указанный тег выходит за пределы диапазона, определяемого граничными значениями. С этой целью в WinCC flexible создаются аналоговые сообщения
- Процедура нумерованного сообщения  
Номер сообщения (и соответствующий текст сообщения) передается из ПЛК на устройство HMI. В этом случае сообщения конфигурируются при программировании ПЛК:
  - В SIMATIC STEP 7:  
ALARM\_S сообщения
  - В SIMOTION SCOUT:  
ALARM\_S сообщения и технологические сообщения

##### Квитирование сообщений

Для сообщений, информирующих о критических или опасных рабочих состояниях или состояниях процесса, можно запрограммировать условие, в соответствии с которым, оператор должен будет квитировать эти сообщения.

##### Состояния сообщения

Дискретные и аналоговые сообщения могут находиться в следующих состояниях:

- При возникновении условия, генерирующего сообщение, это сообщение находится в состоянии "Activated [Активно]". Как только сообщение квитировано оператором, его состояние принимает значение "Activated/acknowledged [Активно/квитировано]".
- При исчезновении условия, вызывающего появление сообщения, сообщение переходит в состояние "Activated/deactivated [Активно/ неактивно]". Как только неактивное сообщение квитировано оператором, его состояние принимает значение "Activated/deactivated/acknowledged [Активно/ неактивно/ квитировано]".

Информация о переходе сообщения в каждое из описанных состояний может быть отображена на экране устройства, занесена в журнал или выведена на печать.

### 6.1.2.2 Квитирование сообщений

#### Введение

Для дискретных и аналоговых сообщений, информирующих о критических или опасных рабочих состояниях или состояниях процесса, можно запрограммировать условие, в соответствии с которым, оператор должен будет квитировать эти сообщения.

#### Механизм квитирования сообщений

Сообщения могут быть квитированы либо оператором с устройства HMI либо программой управления ПЛК. При квитировании сообщения на устройстве HMI оператором в теге ПЛК устанавливается определенный бит.

В первом случае оператор может использовать один из следующих вариантов:

- Клавиша квитирования <АСК> (возможно только на некоторых устройствах HMI)
- Функциональные клавиши, программируемые клавиши или кнопки на экранах

Кроме того, сообщения могут быть квитированы с помощью системных функций, вызываемых в списках функций или скриптов.

#### Сообщения, требующие квитирования

Необходимость квитирования сообщения определяется классом сообщения.

Классы сообщений главным образом определяют вид отображения сообщения на устройстве HMI, а также процедуру квитирования. В WinCC flexible предоставляется возможность использования уже созданных классов сообщений, а также возможность создания пользовательских классов сообщений.

#### Квитирование в ПЛК

Дискретное сообщение может быть квитировано в ПЛК путем установки определенного бита в теге ПЛК.

### Групповое квитирование сообщений

При конфигурировании сообщений пользователь может указать, будет ли оператор квитировать каждое сообщение по отдельности, или сообщения одной и той же группы сообщений могут быть квитированы все вместе. Имеет смысл объединять в группы сообщения, вызванные одной и той же ошибкой.

#### 6.1.2.3 Классы сообщений

##### Классы сообщений

Классы сообщений определяют, главным образом, вид отображения сообщений на устройстве HMI. Кроме того, классы сообщений используются для того, чтобы группировать сообщения для отображения с использованием различных средств.

В WinCC flexible предоставляет возможность использования уже созданных классов сообщений, а также возможность создания пользовательских классов сообщений.

##### Возможные параметры настройки для класса сообщений

Для каждого класса сообщений могут быть определены следующие параметры настройки:

- Квитирование: Все сообщения этого класса должны быть квитированы.
- Тексты, цвета и режимы мигания для указания состояний сообщений при их отображении
- Журнал сообщений для архивирования всех событий, связанных с изменением состояния сообщений в данном классе.
- Текст, помещаемый перед номером сообщения для указания класса сообщения при его отображении на устройстве HMI.
- Адрес электронной почты, по которому будут отправляться все сообщения о событиях, связанных с изменением состояния сообщений данного класса.

##### Предварительно определенные классы сообщений в WinCC flexible

- "Error [Сообщения об ошибках]" для дискретных и аналоговых сообщений, которые указывают на критические или опасные рабочие состояния системы управления или состояния технологического процесса. Сообщения этого класса всегда должны квитироваться.
- "Operation [Оперативные сообщения]" для дискретных и аналоговых сообщений, которые информируют о нормальных рабочих состояниях системы управления, состояниях технологического процесса и технологических операциях. Сообщения этого класса не требуют квитирования.
- "System [Системные сообщения]" для системных сообщений, которые уведомляют оператора о рабочих состояниях устройства HMI и контроллеров. Этот класс сообщений нельзя использовать для пользовательских сообщений.

При редактировании этих предварительно определенных классов сообщений можно менять только некоторые определенные свойства.

### 6.1.3 Системные сообщения

#### Введение

Системные сообщения информируют оператора о рабочих состояниях устройства HMI и контроллеров. Характер системных сообщений может меняться от сообщения-замечания до сообщения о серьезной ошибке.

#### Вывод системных сообщений

Устройство HMI или ПЛК генерирует сообщение, если на одном из этих устройств возникает ошибка или одно из этих устройств переходит в определенное состояние, либо возникает ошибка или определенное состояние во время обмена данными между этими устройствами.

Системное сообщение состоит из номера и текста сообщения. В тексте сообщения содержится также значение внутреннего системного тега, которое более точно указывает причину сообщения. Для системных сообщений можно задать только определенные свойства.

#### Типы системных сообщений

Существуют два типа системных сообщений:

- Системные сообщения HMI  
Системные сообщения HMI генерируются устройством HMI при возникновении ошибок или определенных состояний устройства, или при возникновении ошибок при обмене данными с ПЛК.
- Системные сообщения ПЛК  
Эти сообщения генерируются ПЛК и не могут быть сконфигурированы в WinCC flexible.

#### Отображение сообщений на устройстве HMI

В основных параметрах настройки системы сообщений определяется тип отображения системных сообщений на устройстве HMI, а также время отображения системного сообщения.

Для отображения системных сообщений на устройстве HMI используйте объекты "alarm view [окно просмотра сообщений]" и "alarm window [окно сообщений]".

Всякий раз при конфигурировании одного из этих объектов на экране или шаблоне выбирайте параметр "System".

#### Системные сообщения конкретного устройства

Руководство по эксплуатации устройства HMI содержит список возможных системных сообщений, причин их возникновения и возможных методов устранения.

Если вы обратитесь в интерактивную службу поддержки по поводу возникновения системного сообщения HMI, вам необходимо будет сообщить номер сообщения и значения тегов, содержащиеся в сообщении.

## 6.1.4 Отображение сообщений

### 6.1.4.1 Отображение сообщений на устройстве HMI

#### Виды отображения сообщений на устройстве HMI

В WinCC flexible предлагаются следующие виды отображения сообщений на устройстве HMI:

- **Окно просмотра сообщений (англ. alarm view)**  
Окно просмотра сообщений настраивается для определенного конкретного экрана. В зависимости от заданных размеров сообщений в окне одновременно может выводиться более одного сообщения. Для различных классов сообщений и различных экранов можно сконфигурировать более одного окна просмотра сообщений.  
Окно просмотра сообщений может быть сконфигурировано таким образом, что оно будет содержать только одну строку сообщения.
- **Окно сообщений (англ. alarm window)**  
Окно сообщений конфигурируется на шаблоне экрана и поэтому является компонентом всех экранов в проекте. В зависимости от заданных размеров сообщений в окне одновременно может выводиться более одного сообщения. Открытие и закрытие окна сообщений может быть вызвано определенным событием. По практическим соображениям окна сообщений размещаются в отдельном слое, что позволяет скрыть их в процессе разработки проекта.

#### Дополнительная сигнализация: индикатор сообщений

Индикатор сообщений представляет собой графический символ, отображаемый на экране при возникновении сообщения. Индикатор сообщений конфигурируется на шаблоне экрана и поэтому является компонентом всех экранов в проекте.

Индикатор сообщений может находиться в одном из двух состояний:

- **Мигание:** По крайней мере, одно сообщение ожидает квитирования.
- **Статическое состояние:** Сообщения квитированы, но, по крайней мере, одно из них еще активно.

Реакцией устройства HMI на сообщение может быть запуск функций списка, определенного при проектировании

### 6.1.4.2 Архивирование и вывод сообщений на печать

#### Анализ и документирование сообщений

В дополнение к отображению сообщений в режиме реального времени с использованием "окна просмотра сообщений" и "окна сообщений" WinCC flexible предоставляет следующие возможности для анализа и документирования сообщений:

- Информация о событиях сообщений может быть распечатана сразу же при возникновении сообщения.
- Информация о событиях сообщений из буфера сообщений может быть распечатана в виде отчетов.
- Информация о событиях сообщений может быть занесена в журнал сообщений.
- Информация о событиях сообщений из журнала сообщений может быть отображена на устройстве HMI или распечатана в виде отчета.

### Немедленная печать сообщений

В основных параметрах настройки системы сообщений вы можете включить или отключить функцию печати сообщений всего проекта. Кроме того, можно включить функцию печати каждого отдельного сообщения.

### Занесение сообщений в журнал сообщений

Использование классов сообщений позволяет определить журнал сообщений для архивирования указанной группы сообщений. Журнал сообщений может быть определен для каждого класса сообщений. Информация обо всех событиях, относящихся к сообщениям некоторого класса заносится в определенный для данного класса журнал сообщений.

### Вывод сообщений в виде отчета

Путем задания соответствующих свойств объекта "Print alarm [Печать сообщения]" можно указать сообщения, которые будут выводиться в отчете сообщений. При выводе сообщений в отчете в дополнение к фильтрации сообщений из источника данных (буфер сообщений или журнал сообщений) возможны фильтрация сообщений на основе использования классов сообщений.

#### 6.1.4.3 Системные функции для редактирования классов сообщений

##### Системные функции

Системные функции представляют собой готовые функции, которые могут использоваться для решения ряда задач, возникающих во время исполнения проекта, даже теми пользователями, которые не имеют навыков программирования. Системные функции могут использоваться в виде списка функций или вызываться в скрипте.

В приведенной ниже таблице приведены системные функции, предназначенные для редактирования сообщений и управления их отображением.

Системная функция	Действие
EditAlarm	Вызывает функцию "Edit [Правка]" для всех выделенных сообщений.
ClearAlarmBuffer	Удаляет сообщения из буфера сообщений на устройстве HMI.

Системная функция	Действие
ClearAlarmBufferProTool	Аналогично действию функции "ClearAlarmBuffer". Эта системная функция была сохранена для обеспечения совместимости с проектами, разработанными со старыми версиями ProTool.
AlarmViewEditAlarm	Вызывает функцию "Edit [Правка]" для сообщений, выделенных в указанном окне просмотра сообщений.
AlarmViewAcknowledgeAlarm	Квитирует сообщения, выделенные в указанном окне просмотра сообщений.
AlarmViewShowOperatorNotes	Отображает комментарии оператора для сообщения, выделенного в указанном окне просмотра сообщений.
AcknowledgeAlarm	Квитирует все выделенные сообщения.
SetAlarmReportMode	Включает или отключает автоматический вывод сообщений на печать.
ShowAlarmWindow	Скрывает или выводит окно сообщений на экран устройства HMI.
ShowSystemAlarm	Отображает значение параметра в качестве системного сообщения на устройстве HMI.

Более подробное описание этих системных функций можно найти в разделе "Работа с WinCC flexible > Справочная система > Системные функции".

### События сообщений и объекты для отображения сообщений

Во время исполнения проекта для сообщений и объектов, используемых для их отображения на экране, могут быть сгенерированы события, приведенные в следующей таблице. При этом для каждого из этих событий можно определить функцию.

Объект	Возможные события
Дискретное сообщение	Activate [Активизация] Deactivate [Деактивизация] Acknowledgement [Квитирование] Edit [Правка]
Аналоговое сообщение	Activate [Активизация] Deactivate [Деактивизация] Acknowledgement [Квитирование] Edit [Правка]
Окно просмотра сообщений	Enable [Показать] Disable [Скрыть]
Окно сообщений	Enable [Показать] Disable [Скрыть]
Индикатор сообщений	Click [Щелчек кнопкой мыши] Click when flashing [Щелчек при мигании]

Более подробное описание этих событий можно найти в разделе "Работа с WinCC flexible > Справочная система > Системные функции".

## 6.2 Элементы пользовательского интерфейса и основные параметры настройки

### 6.2.1 Элементы и свойства сообщений

#### Свойства сообщений

Сообщение всегда состоит из следующих элементов:

- Текст сообщения  
Текст сообщения представляет собой собственно информацию сообщения. Форматы символов, поддерживаемые соответствующим устройством HMI, можно использовать для посимвольного форматирования текста.  
Текст сообщения может содержать поля вывода текущих значений тегов или текстовых списков. Текущее значение тега в момент изменения состояния сообщения сохраняется в буфере сообщений.
- Номер сообщения  
Номер сообщения используется для ссылок на это сообщение. Номера сообщений уникальны в пределах сообщений одного типа. Различают следующие типы сообщений:
  - Дискретные сообщения
  - Аналоговые сообщения
  - Системные сообщения HMI
  - Сообщения, генерируемые процессором ПЛК
- Триггер сообщения
  - Для дискретных сообщений: бит тега
  - Для аналоговых сообщений: граничное значение для тега
- Класс сообщения  
Класс сообщения определяет, должно ли данное сообщение быть квитировано. Класс сообщения также определяет вид отображения сообщения на устройстве HMI. Кроме того, класс сообщения определяет, будет ли сообщение заноситься в соответствующий журнал сообщений, и в какой журнал это сообщение будет заноситься

---

#### Замечание

Если вы хотите интегрировать проект в среду SIMATIC STEP 7, вы можете создать до 7 классов сообщений в WinCC flexible и STEP 7.

---

Для каждого сообщения эти элементы легко выбираются из предложенных вариантов или вводятся пользователем.

### Дополнительные свойства сообщений

На работу с сообщениями оказывают влияние также следующие их свойства:

- **Группа сообщений**  
Если сообщение относится к определенной группе сообщений, оно может быть квитировано одним действием вместе с другими сообщениями этой же группы.
- **Текст пояснения (англ. infotext)**  
Текст пояснения содержит дополнительную информацию, связанную с сообщением. Текст пояснения отображается в отдельном окне на экране операторского устройства при нажатии оператором кнопки <HELP>.
- **Автоматическое протоколирование сообщений**  
В дополнение к возможности включения и отключения функции вывода сообщений всего проекта в виде отчетов можно включить функцию протоколирования для каждого отдельного сообщения.
- **Квитирование в ПЛК через "тег квитирования в ПЛК" (англ. "Acknowledgement write tag")**  
Дискретное сообщение может быть квитировано программой ПЛК путем установки определенного бита в теге.
- **Отправка информации о квитировании в тег ПЛК - "тег квитирования на HMI" (англ. "Acknowledgement read teg")** При квитировании дискретного сообщения оператором на HMI в теге ПЛК устанавливается определенный бит.

## 6.2.2 Редакторы сообщений

### 6.2.2.1 Основные принципы работы с редакторами

#### Редакторы для создания и редактирования сообщений

В системе WinCC flexible предоставляются следующие табличные редакторы для создания и определения свойств сообщений:

- "Discrete alarms [Дискретные сообщения]" для создания и редактирования свойств дискретных сообщений
- "Analog alarms [Аналоговые сообщения]" для создания и редактирования свойств аналоговых сообщений
- "System alarms [Системные сообщения]" для редактирования текстов системных сообщений
- "Alarm class [Классы сообщений]" для создания и редактирования классов сообщений
- "Alarm groups [Группы сообщений]" для создания и редактирования групп сообщений

### Изменение отображения столбцов

Можно выполнить следующую настройку отображения столбцов таблицы:

- С помощью контекстного меню для заголовка столбца (щелчок правой кнопкой мыши) вы можете скрыть или, наоборот, отобразить отдельные столбцы.  
Этой функции нет в редакторе групп сообщений "Alarm groups", поскольку в этом редакторе отображается только два столбца.
- Вы можете изменить ширину столбца, перетаскивая буксировкой правую границу заголовка столбца.
- Перетаскиванием заголовка столбца вы можете изменить порядок столбцов.  
Этой функции нет в редакторе "Группы сообщений".
- Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца. Щелкните на том же заголовке еще раз для того, чтобы поменять порядок сортировки на противоположный.  
Соответствующий заголовок столбца помечен стрелкой. Направление стрелки указывает на порядок сортировки.

### Удаление и копирование объектов

Один или более объектов можно удалить или скопировать, выделяя всю строку таблицы для каждого объекта с помощью пиктограммы слева от строки.

### Автоматическое заполнение нескольких строк таблицы с помощью функции перетаскивания

В табличных редакторах WinCC flexible существует возможность заполнения нескольких строк таблицы с помощью одной операции.

- Создание нескольких новых объектов (сообщений, классов сообщений или групп сообщений) с одинаковыми свойствами:
  - Выполните сортировку таблицы таким образом, чтобы копируемая строка была помещена в конце таблицы.
  - Выделите первый копируемый элемент в строке.

- Держа нажатой левую кнопку мыши, перетащите правый нижний угол выделенного элемента таблицы вниз, в пустую часть таблицы.
- Перенос свойств нескольким объектам (например, изменение тегов-триггеров):
  - Выделите элемент таблицы с соответствующим свойством.
  - Держа нажатой левую кнопку мыши, перетащите правый нижний угол выделенного элемента таблицы вниз, в строки таблицы, которые необходимо изменить.

### **Функция перетаскивания в пределах таблицы**

Функцию перетаскивания можно использовать для копирования отдельного свойства (например, текста сообщения или цвета сообщения) одного из элементов таблицы другому элементу таблицы.

### **Перетаскивание из окна объектов.**

Операцию перетаскивания можно использовать для перемещения объекта (например, тега) из окна объектов в ячейку таблицы, при условии, что данный объект допустим в этой ячейке таблицы.

## **6.2.2.2 Редактор дискретных сообщений**

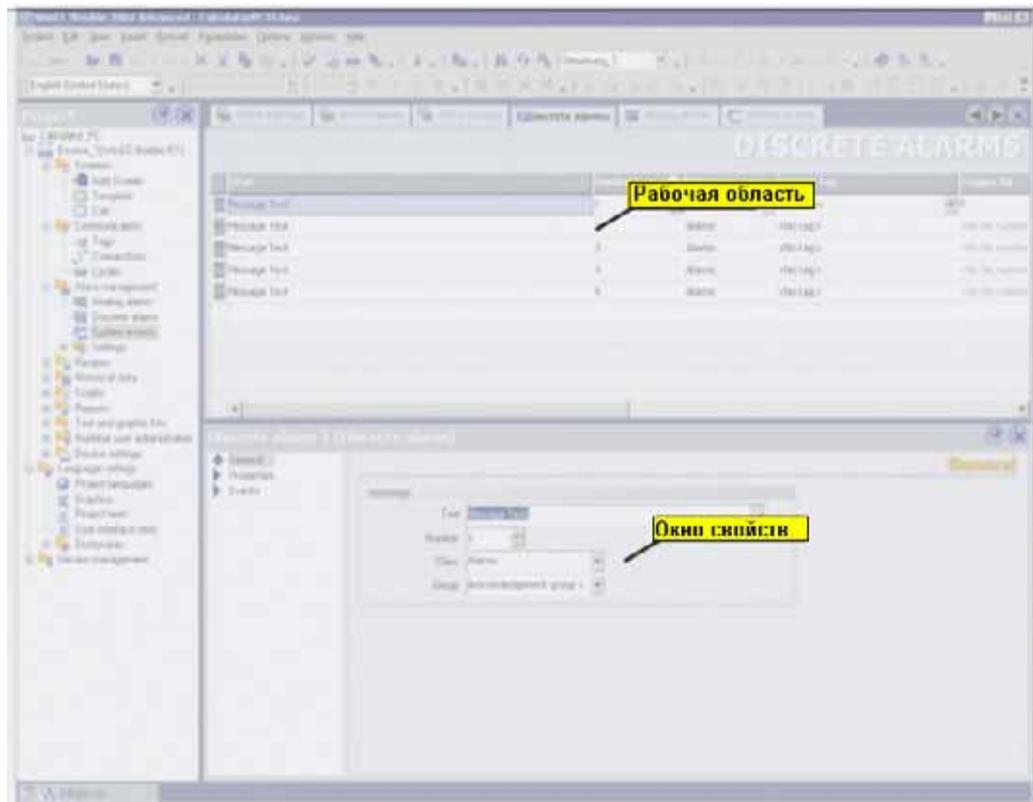
### **Введение**

В редакторе "Discrete alarms [Дискретные сообщения]" создаются дискретные сообщения и определяются их свойства.

### **Открытие редактора**

Щелкните дважды на элементе "Discrete alarms [Дискретные сообщения]" в группе "Alarms[Сообщения]", отображаемой в окне проекта.

## Пользовательский интерфейс редактора



### Рабочая область

Все дискретные сообщения отображаются в таблице в рабочей области. Свойства дискретных сообщений можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

### Окно свойств

В окне свойств определяются свойства дискретных сообщений. В окне свойств отображается та же информация и параметры, что и в таблице рабочей области.

### 6.2.2.3 Редактор аналоговых сообщений

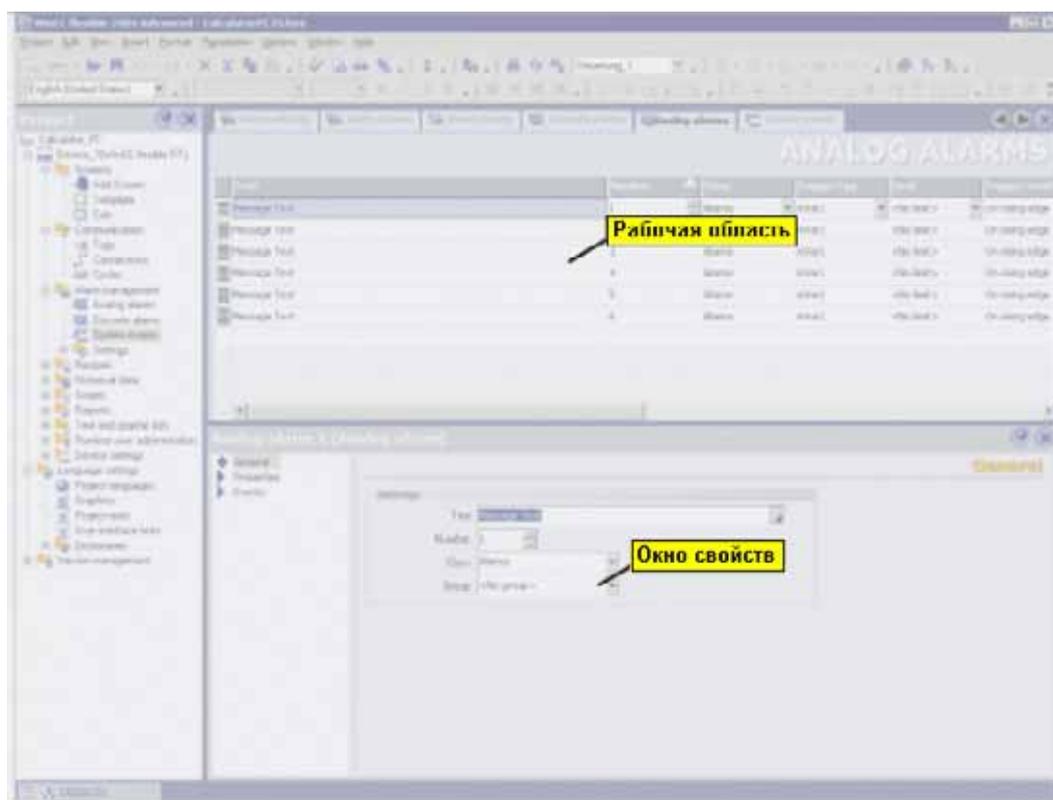
#### Введение

В редакторе "Analog alarms [Аналоговые сообщения]" создаются аналоговые сообщения и определяются их свойства.

## Открытие редактора

Щелкните дважды на элементе "Analog alarms [Аналоговые сообщения]" в группе "Alarms[Сообщения]", отображаемой в окне проекта.

## Пользовательский интерфейс редактора



### Рабочая область

Все аналоговые сообщения отображаются в таблице в рабочей области. Свойства аналоговых сообщений можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

### Окно свойств

В окне свойств определяются свойства аналоговых сообщений. В окне свойств отображается та же информация и параметры, что и в таблице рабочей области.

#### 6.2.2.4 Редактор системных сообщений

##### Введение

Просмотреть системные сообщения HMI и изменить текст системных сообщений можно в табличном редакторе системных сообщений.

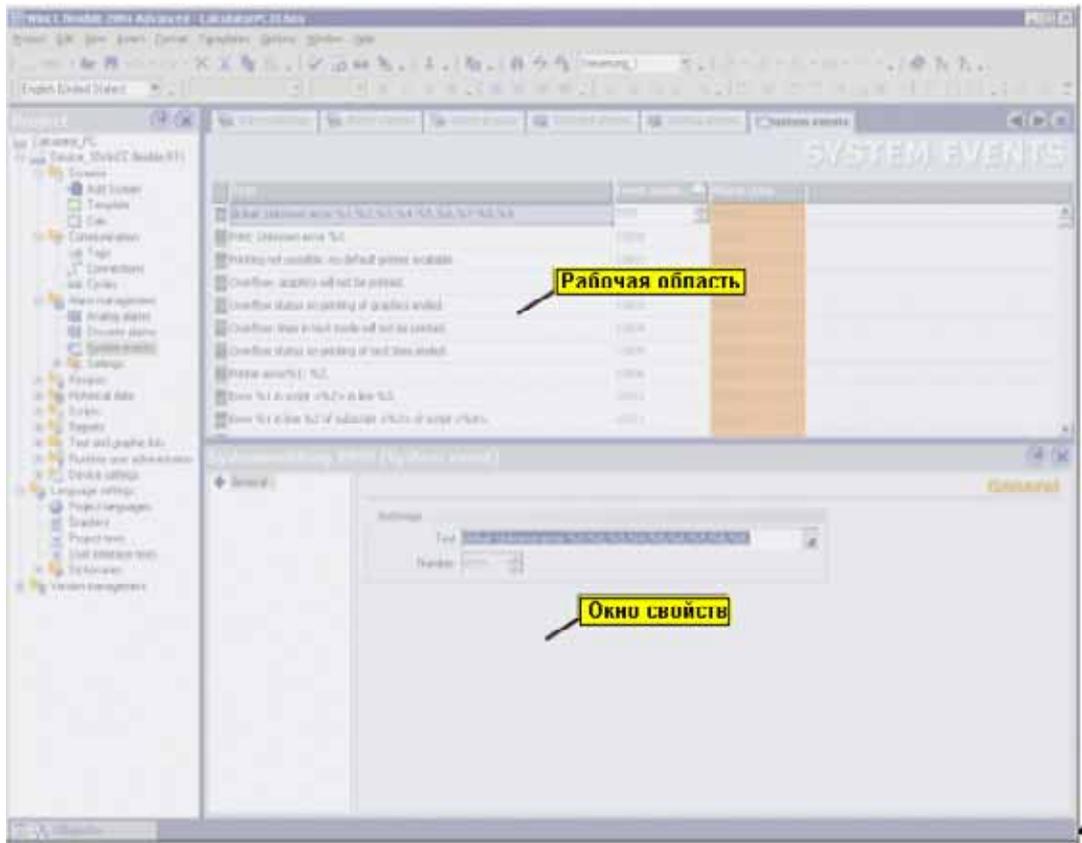
##### Открытие редактора системных сообщений

Щелкните дважды на элементе "System alarms [Системные сообщения]" в группе "Alarms[Сообщения]", отображаемой в окне проекта.

По умолчанию элемент "System alarms [Системные сообщения]" не отображается в окне проекта WinCC flexible. Для его отображения выполните следующие действия:

1. Выберите команду "Settings [Настройка]" в меню "Options [Сервис]".
2. В диалоговом окне "Setting [Настройка]" откройте категорию параметров настройки "Workbench > Settings for project view [Рабочее место > Настройка окна проекта]".
3. В поле "Change the mode in which the project tree is shown [Изменить режим отображения иерархической структуры проекта]" выберите опцию "Display all items [Отображать все элементы]".

### Пользовательский интерфейс редактора



#### Рабочая область

Все системные сообщения отображаются в таблице в рабочей области. Тексты системных сообщений можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

#### Окно свойств

Изменение текстов системных сообщений выполняется в окне свойств. Номер сообщения и класс сообщения присваиваются сообщению системой.

#### 6.2.2.5 Редактор классов сообщений

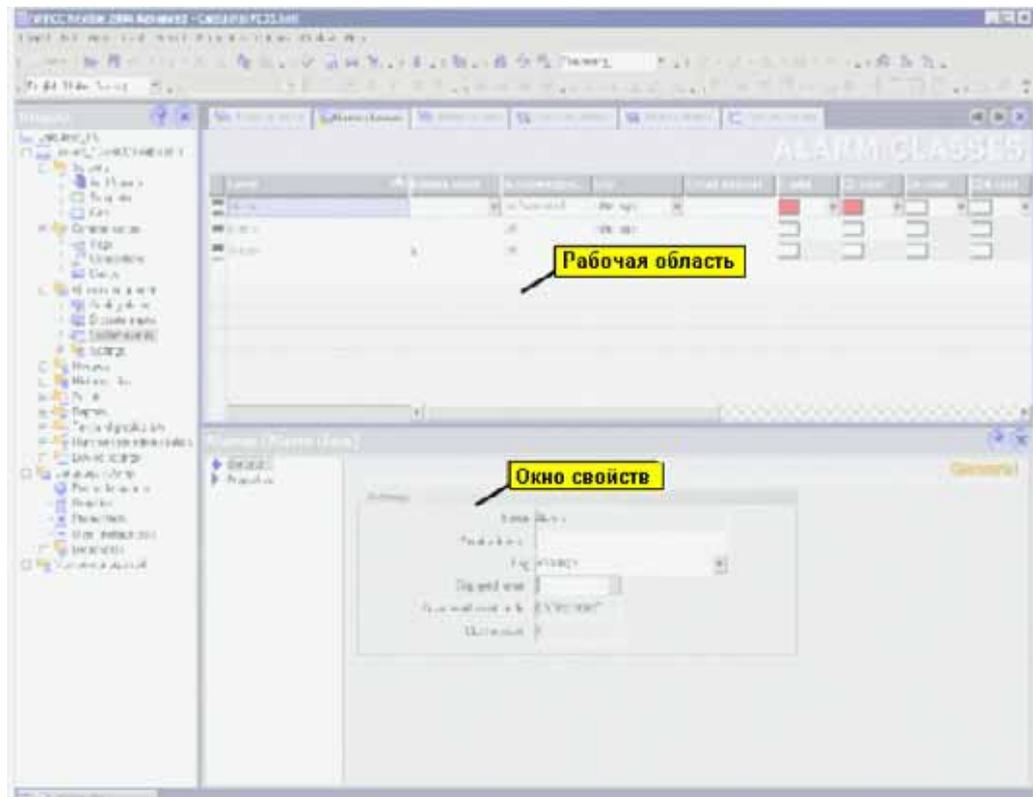
##### Введение

В табличном редакторе "Alarm classes [Классы сообщений]" создаются классы сообщений и определяются их свойства.

### Открытие редактора классов сообщений

Щелкните дважды на элементе "Alarm classes [Классы сообщений]" в группе "Alarms[Сообщения]", отображаемой в окне проекта.

### Пользовательский интерфейс редактора



#### Рабочая область

Все классы сообщений отображаются в таблице в рабочей области. Свойства классов сообщений можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

#### Окно свойств

В окне свойств определяются свойства классов сообщений. В окне свойств отображается та же информация и параметры, что и в таблице рабочей области.

### 6.2.2.6 Редактор групп сообщений

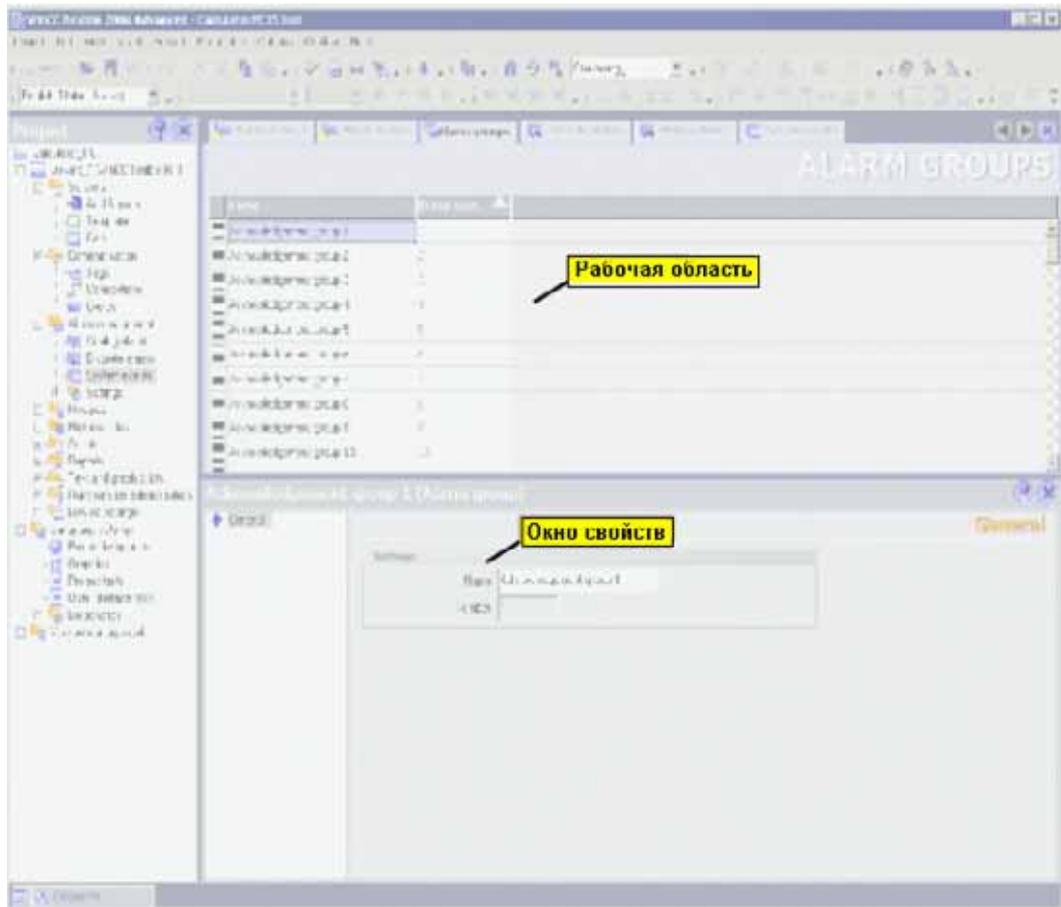
#### Введение

В табличном редакторе "Alarm groups [Группы сообщений]" создаются группы сообщений и определяются их свойства.

### Открытие редактора групп сообщений

Щелкните дважды на элементе "Alarm groups [Группы сообщений]" в группе "Alarms[Сообщения]", отображаемой в окне проекта.

### Пользовательский интерфейс редактора



#### Рабочая область

Все группы сообщений отображаются в таблице в рабочей области. Свойства групп сообщений можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

#### Окно свойств

Имя группы сообщений можно изменить в окне свойств. Номер назначается системой.

### 6.2.3 Основные параметры настройки системы сообщений

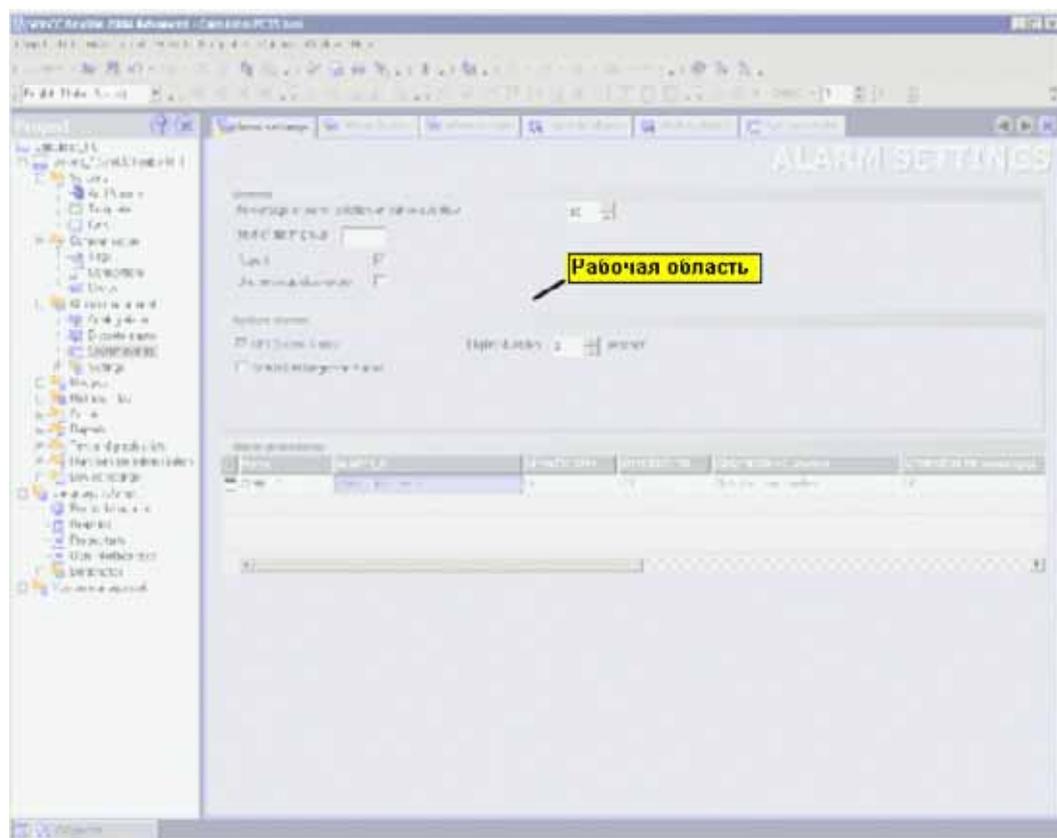
#### Введение

Система сообщений WinCC flexible может функционировать с параметрами настройки по умолчанию. Вам необходимо изменить эти параметры настройки по умолчанию в том случае, если вы хотите адаптировать систему сообщений и работу системы сообщений к конкретным условиям функционирования установки.

#### Открытие окна настройки системы сообщений

Щелкните дважды на элементе "Alarm settings [Настройка системы сообщений]" в группе "Alarms > Settings [Сообщения > Настройка]".

#### Пользовательский интерфейс редактора



#### Рабочая область

Параметры настройки системы сообщений определяются в рабочей области. В области "System alarms [Системные сообщения]" выбираются, например, типы системных сообщений, которые будут отображаться на устройстве HMI. Дополнительные параметры, требуемые при интеграции проекта, определяются в области "Alarm procedures [Процедуры сообщений]".

## 6.3 Использование системы сообщений

### 6.3.1 Вывод сообщений в виде отчета

#### Введение

WinCC flexible позволяет выводить сообщения из буфера сообщений или журнала сообщений в виде отчета.

#### Данные, выводимые в отчете сообщений

Для того чтобы вывести сообщения из буфера сообщений или журнала сообщений в виде отчета, вставьте объект "Print alarm [Печать сообщения]" из окна инструментов в отчет. Выделите объект с тем, чтобы свойства объекта отобразились в окне свойств. В окне свойств определите данные для вывода в отчете.

В отчете могут быть выведены следующие данные:

- Текущие сообщения из буфера сообщений
- Сообщения из журнала сообщений

Определите классы, сообщения которых вы хотите вывести. При этом возможны следующие случаи:

- Error [Сообщения об ошибках]
- Operation [Оперативные сообщения]
- Control [Управление]

Определите последовательность вывода сообщений.

При этом возможны следующие случаи:

- Первым выводится самое старое сообщение
- Первым выводится самое последнее сообщение

Для вывода сообщений за определенный период свяжите поля "Display beginning [Начало вывода]" и "Display end [Конец вывода]" с тегами. В режиме исполнения в тегах будет храниться дата и время вывода первого и последнего сообщения, сгенерированных в указанный период.

### 6.3.2 Использование сообщений из STEP7 с процедурами нумерованных сообщений

#### Конфигурирование сообщений в SIMATIC STEP 7

ALARM\_S и ALARM\_D представляют собой процедуры вывода нумерованных сообщений. Номера сообщений определяются автоматически в процессе программирования в STEP 7. Эти номера уникальны и используются для идентификации сообщений.

В процессе настройки системы сообщений в STEP 7 сообщения и их атрибуты объединяются с проектными данными STEP 7. WinCC flexible автоматически импортирует требуемые данные и затем передает их на устройство HMI.

WinCC flexible позволяет отфильтровать отображаемые сообщения ALARM\_S, используя классы отображения. Щелкните дважды на элементе "Alarm settings [Настройка системы сообщений]" в группе "Alarms > Settings [Сообщения > Настройка]". Существующие соединения отображаются в области "Alarm Procedures [Процедуры сообщений]".



В строке требуемого соединения выделите поле в столбце "ALARM\_S Display Classes [Классы отображения ALARM\_S]" и откройте диалоговое окно выбора, нажав кнопку выбора. Выберите требуемый класс отображения. Закройте диалоговое окно, нажав кнопку .

В столбце "SFM Alarms [Сообщения SFM]" соединения определите, будет ли отображаться системная ошибка. Более подробную информацию можно найти в документации STEP 7.

### Отображение классов сообщений

Сообщения ALARM\_S и ALARM\_D относятся к определенным классам сообщений в STEP 7. Для редактирования параметров отображения для этих классов сообщений выберите "Alarms > Settings > Alarm Classes [Сообщения > Настройка > Классы сообщений]". Откройте контекстное меню и выберите команду "Open Editor [Открытие редактора]". Идентифицировать сообщения этого класса можно по префиксу S7 в имени класса сообщений.

Name	Acknowledgment	Log	I color	IO color	IA color	I/OA color
S7Alarm	On "incoming"	<No log>	Red	Orange	Yellow	Green
S7NoAlarm	off	<No log>	Magenta	Cyan		
S7OperationMessage	off	<No log>				
S7OperatorInputRequest	off	<No log>				
S7ProcessControlMaintenance	off	<No log>				
S7ProcessControlSystemMessageOs	off	<No log>				
S7ProcessControlSystemMessagePlc	off	<No log>				
S7ProcessMessageAlarm	On "incoming"	<No log>				
S7ProcessMessageEvent	off	<No log>				
S7StatusMessage	On "incoming"	<No log>				
S7Tolerance	off	<No log>				
S7Warning	off	<No log>				

Параметры отображения классов сообщений определяются в редакторе "Alarm Classes [Классы сообщений]".

## 6.4 Архивирование сообщений

### 6.4.1 Основные сведения об архивировании сообщений

#### Введение

Сообщения информируют об ошибочных и рабочих состояниях технологического процесса в проекте. Эти сообщения обычно генерируются контроллером. Сообщения выводятся на экран устройства HMI.

WinCC flexible предоставляет возможность архивировать сообщения и документировать данные о рабочих и ошибочных состояниях технологического процесса.

Не все устройства HMI позволяют архивировать сообщения.

#### Основные сведения

WinCC flexible позволяет настроить процесс записи сообщений в журнал – архивирование сообщений. Сообщения записываются в журнал сообщений в соответствии с классом сообщений, которому они относятся. Каждое сообщение

относится к определенному классу сообщений. При определении параметров и свойств класса сообщений необходимо ввести имя журнала сообщений, используемого для данного класса.

Вы можете архивировать в одном журнале сообщения разных классов сообщений.

При создании журнала определяются свойства журнала и режим его функционирования.

В журнале содержатся следующие данные:

- Дата и время сообщения
- Тест сообщения
- Номер сообщения
- Состояние сообщения
- Класс сообщения
- Процедура сообщения
- Значения тегов, содержащихся в тексте сообщения
- Информация о контроллере

---

**Замечание**

Текст сообщения и информация о контроллере заносится в журнал только в том случае, если это было указано в свойствах журнала.

---

## 6.4.2 Основные сведения о журналах сообщений

### Введение

Для занесения сообщений в соответствующие журналы, они группируются по классам сообщений. Сообщения каждого класса могут заноситься в отдельный журнал сообщений. В зависимости от устройства HMI при настройке процесса архивирования можно выбрать один из нескольких типов журналов. Свойства журнала определяют функционирование журнала во время записи в него сообщений.

### Типы журналов

WinCC flexible позволяет выбрать один из следующих типов журналов:

1. Циклический журнал
2. Сегментированный циклический журнал
3. Журнал с уровневой системой сообщений
4. Журнал с исполнением системных функций в зависимости от уровня

Сообщения могут архивироваться автоматически или под управлением оператора.

### Среда для хранения данных журнала

Данные журнала могут храниться либо в файле, либо в базе данных. Сохраненные данные могут быть обработаны другими программами, например, с целью их анализа.

### Отображение содержимого журнала

Содержимое журнала может быть отображено на устройстве HMI. С этой целью необходимо сконфигурировать окно просмотра сообщений.

## 6.4.3 Редактор журналов сообщений

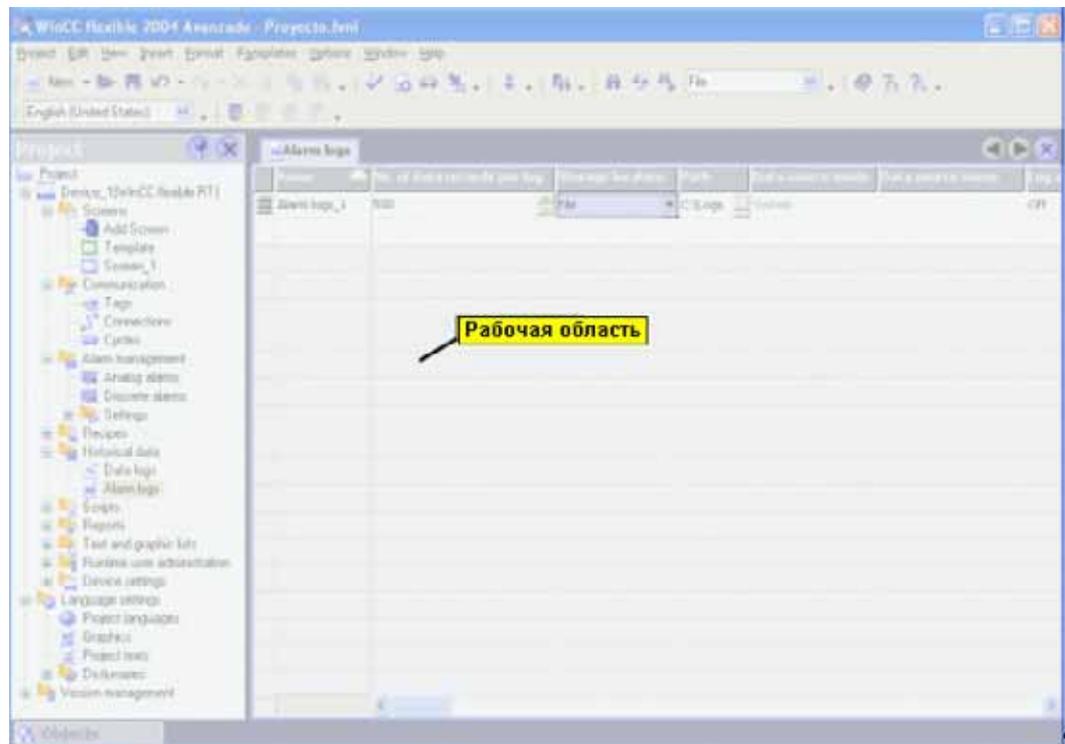
### Введение

В табличном редакторе "Alarm logs [Журналы сообщений]" создаются журналы сообщений для архивирования сообщений определенных классов и определяются их свойства.

### Открытие редактора журналов сообщений

Для открытия редактора журналов сообщений дважды щелкните на элементе "Alarm logs [Журналы сообщений]" в группе "Log [журнал]" окна проекта.

## Пользовательский интерфейс редактора



### Рабочая область

Все журналы сообщений отображаются в таблице в рабочей области. Свойства журналов сообщений можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

## 6.4.4 Определение основных свойств журналов сообщений

### Введение

Свойства журналов сообщений могут быть определены в редакторе журналов сообщений или в окне свойств журналов.

### Общие свойства

- Name [Имя]

Имя журнала сообщений задается произвольным образом, однако имя должно включать по крайней мере одну букву или одну цифру.

	<p><b>Замечание</b></p> <p>Символы, которые могут использоваться в имени источника данных, зависят от места хранения.</p> <p>Если в качестве места хранения используется "File [Файл]", можно использовать следующие символы: \ / * ? : " &lt; &gt;  </p> <p>Если в качестве места хранения используется "Database [База данных]", можно использовать следующие символы: a-z A-Z 0-9 _ @ # \$</p> <p>Однако, символы _ @ # \$ нельзя использовать в качестве первого символа имени.</p>
--	---

- Область памяти для хранения журнала
 

Журнал сообщений может храниться в базе данных ODBC (только на ПК) или в отдельном "\*.csv"-файле. Выберите в качестве места хранения "File [Файл]" или "Database [База данных]" соответственно.

Для определения пути к месту хранения ("path") вы можете, в зависимости от устройства HMI выбрать жесткий диск ПК или карту памяти панели, или, если возможно, сетевой дисковый накопитель.

Если в качестве места хранения выбрана база данных ODBC, вы можете использовать имя, предлагаемое системой (системное имя источника данных) или ввести свое имя (пользовательское имя источника данных).
- Size [Размер]
 

Размер журнала определяется исходя из количества записей данных и приблизительного размера записи. Длина записи журнала зависит кроме всего прочего от того, заносится ли вместе с сообщением в журнал текст сообщения и значения соответствующих тегов.

**Свойства журнала сообщений**

- Запуск процедуры архивирования
 

С помощью соответствующих свойств можно настроить процедуру записи сообщений в журнал таким образом, что запись сообщений начнется при запуске системы исполнения проекта. С этой целью установите флажок опции "Enable logging at runtime start [Начать архивирование при запуске среды исполнения]"

Вы можете настроить функционирование журнала сообщений в режиме исполнения другими способами. Установите флажок опции "Reset log [Очистить журнал]", если вы хотите заменить данные, записанные в журнал ранее, новыми данными или флажок опции "Append data to existing log [Добавить данные в существующий журнал]", если вы хотите добавить новые данные в уже существующий журнал.

---

	<p><b>Замечание</b></p> <p>Для очистки журнала в режиме исполнения могут быть использованы системные функции.</p>
--	---

---

- Метод архивирования
 

Метод архивирования определяет, что происходит с журналом, когда он полон. Вы можете выбрать один из следующих вариантов:

- Циклический журнал: При заполнении журнала новые данные записываются на место самых старых записей.
- Сегментированный циклический журнал: Создается несколько журналов одного и того же размера - сегментов. Журналы заполняются данными. Когда полностью заполнены все журналы, новые данные заносятся в журнал, который был заполнен первым.
- Вывод системного сообщения: при достижении определенного уровня заполнения появляется системное сообщение.
- Генерирование события: Как только журнал заполнен, генерируется событие Overflow [Переполнение].
- **Настройка**

Это свойство определяет, будет ли при архивировании сообщения заноситься в журнал также текст сообщения и информация о месте возникновения ошибки. Тексты сообщений заносятся в журнал на языке среды исполнения, установленном в данный момент.
- **Комментарий**

Комментарий содержит описание журнала.

## 6.4.5 Архивирование сообщений

### Введение

При исполнении проекта сообщения могут записываться в журналы с целью их анализа на более позднем этапе. При настройке процедуры архивирования сообщений пользователь определяет журнал, в который будут записываться сообщения. Кроме того, определяется, какая информация будет заноситься в журнал: только события, связанные с сообщениями, или соответствующие тексты сообщений и информация о месте возникновения ошибок.

### Основные сведения

Для настройки архивирования сообщений необходимо выполнить ряд действий:

- **Создание и определение свойств журналов сообщений**

При создании журнала сообщений необходимо определить следующие свойства:

  - Общие свойства, например, имя, размер, область памяти для хранения
  - Функционирование при запуске среды исполнения
  - Функционирование при заполнении журнала
- **Архивирование сообщений, организованных в классы сообщений**

Журнал сообщений может быть определен для каждого класса сообщений. При этом в нем будет храниться информация, связанная с сообщениями и событиями сообщений данного класса, возникающими при исполнении проекта.

- Дальнейшая обработка архивированных сообщений  
Анализ сообщений, занесенных в журналы, можно выполнить непосредственно в проекте WinCC flexible, например, в окне просмотра сообщений или с помощью другого приложения, например, Excel.

### 6.4.6 Отображение журнальных сообщений на экранах

#### Введение

Сообщения из журналов сообщений могут быть отображены на устройстве HMI на экранах среды исполнения. При этом сообщения определенного класса сообщений загружаются из базы данных (журнала) и выводятся в окне просмотра сообщений.

#### Основные сведения

Для отображения журнальных сообщений на устройстве HMI необходимо создать и настроить окно просмотра сообщений. При создании окна просмотра сообщений, необходимо указать класс сообщений, которые будут отображаться.

### 6.4.7 Структура \*.csv файла с сообщениями

#### Введение

В файле формата \*.csv (Comma separated value), столбцы таблицы (имя и значение записи) отделяются точкой с запятой. Каждая строка таблицы заканчивается возвратом каретки.

#### Пример файла формата \*.csv

В следующем примере показан файл с сообщениями:

```
"Time_ms";"MsgProc";"StateAfter";"MsgClass";"MsgNumber";"Var1";...;"Var8";"TimeString";"MsgText";"PLC"37986550590,27;1;1;3;110001;"";...;"";"30.06.99 13:12:51";"Change to operating mode 'online'";37986550682,87;1;1;3;140010;"";...;"";"30.06.99 13:12:59";"Connection established: PLC_1, Station 2, Rack 0, Position 2";
```

#### Структура файла журнала в формате \*.csv

Следующие значения вводятся в отдельные столбцы журнального файла WinCC flexible:

Параметр	Описание
Time_ms [время_мсек]	Определяет метку времени в виде десятичного значения (по поводу преобразования данных см ниже)
Msg_Proc [процедура_с ообщ]	Процедуры сообщений: 0 = Неизвестная процедура сообщения 1 = Системное сообщение

Параметр	Описание
	2 = Процедура битового сообщения (оперативные сообщения) 3 = Процедура нумерованного сообщения ALARM_S 4 = Диагностическое событие 100 = Процедура битового сообщения (сообщения об ошибках)
State after [состояние после]	События сообщения: 0 = Поступило/Ушло 1 = Поступило 2 = Поступило/Квитировано/Ушло 3 = Поступило/Квитировано 6 = Поступило/Ушло/Квитировано
Msg_Class [класс_сообщ ]	Класс сообщения 0 = Нет классов сообщений 1 = "Сообщения об ошибках" 2 = "Оперативные сообщения" 3 = "Системные сообщения" 64 ... = Пользовательские классы сообщений
Msg Number [номер сообщ]	Номер сообщения
Var1 to Var8 Перем1- Перем8]	Значение тега сообщения, тип данных STRING [СТРОКА]
Time string [строка времени]	Временная метка в виде строки (тип данных STRING), то есть читаемый формат данных
Msg text [текст сообщ]	Сообщение в читаемом формате строки (тип данных STRING)
PLC [ПЛК]	Локализация сообщения (соответствующий ПЛК)

### Преобразование десятичного значения временной метки

Если значение необходимо преобразовать с помощью другой программы, выполните следующие действия:

1. Разделите значение времени в мсек Time\_ms на 1,000,000.

Пример:  $37986476928 : 1.000.000 = 37986,476928$

2. Целая часть числа (37986) это дата, отсчитанная от 31.12.1899.

Теперь вы можете преобразовать значение временной метки в дни в программе Excel путем назначения соответствующих форматов группы "Date [Дата]" ячейкам, содержащим временную метку.

Результат: значение 37986 приводит к значению даты 31.12.2003

3. Дробная часть числа (значение после запятой) - (0,476928) указывает время:
  - Путем умножения значения (0,476928) на 24 получается результат в часах (11,446272).
  - Путем умножения остатка (0,446272) на 60 получается результат в минутах (26,77632).

- Путем умножения остатка (0,77632) на 60 получается результат в секундах (46,5792).

Итог 11:26:46.579

Такое преобразование поддерживается, например, Microsoft Excel.

## 6.4.8 Непосредственный доступ к журнальной базе данных ODBC

### Введение

Журнал может быть организован в виде базы данных или файла.

Обращение к базе данных происходит по ее имени "Data source name (DSN) [Имя источника данных]" Выберите базу данных, которую вы хотели бы использовать в WinCC flexible с помощью меню Пуск, выбрав команды Settings > Control panel > ODBC data sources [Настройка > Панель управления > Источник данных ODBC].

Для сохранения данных журнала укажите при настройке вместо имени каталога имя источника данных - "Data source name" (DSN). С помощью имени источника DSN вы обращаетесь к базе данных и к области памяти, где она хранится.

### Применение

Для дополнительной обработки и анализа данных журнала можно использовать весь спектр возможностей работы с базами данных.

### Основные сведения

Вы создаете источник данных, который связан с базой данных на том же компьютере, на котором содержится программное обеспечение среды исполнения. Имя созданного источника данных DSN указывается при создании журнала в WinCC flexible.

С помощью интерфейса ODBC можно непосредственно обращаться к этой базе данных из других программ, таких как MS Access или MS SQL server.

Используя системную функцию "StartProgram [Запустить программу]", вы можете настроить вызов программы с устройства HMI (например, для приложения MS Access). При этом последовательность вызова программ среды исполнения не нарушается.

## Использование соединений

### 7.1 Основные сведения об использовании соединений

#### 7.1.1 Основы обмена данными

##### Введение

Термин обмен данными относится к обмену данными между двумя коммуникационными партнерами или партнерами по связи. Коммуникационные партнеры могут быть связаны напрямую через кабель или по сети.

##### Коммуникационные партнеры

Коммуникационным партнером может быть любой узел сети, для которого возможно установление соединения и обмен данными с другими узлами сети. В среде WinCC flexible такими узлами могут быть ЦПУ и коммуникационные модули систем автоматизации (контроллеров), устройств HMI и процессорных устройств ПК.

Данные, которыми обмениваются коммуникационные партнеры, могут быть предназначены для решения различных задач. В WinCC flexible, такими задачами могут быть:

## 7.1 Основные сведения об использовании соединений

- управление процессом
- сбор данных процесса
- информирование о состояниях процесса
- архивирование данных процесса

### 7.1.2 Принципы обмена данными

#### Введение

Обмен данными между ПЛК и HMI в WinCC flexible осуществляется с использованием тегов и указателей области.

#### Обмен данными с помощью тегов

Централизованное управление и работа с тегами в WinCC flexible выполняется в редакторе тегов. Существуют внутренние теги и внешние теги. Внешние теги используются для обмена данными с ПЛК и представляют образы определенных областей памяти в ПЛК. И устройство HMI и ПЛК имеют к этим областям памяти доступ на чтение и запись данных. Операции чтения и записи могут быть циклическими или инициироваться событиями.

Создайте в проектных данных теги, которые указывают на определенные адреса в ПЛК. Устройство HMI считывает значения по этим адресам и отображает их. Кроме того, оператор может вводить значения с устройства HMI. Эти значения будут записаны по соответствующим адресам в ПЛК.

#### Обмен данными с помощью указателей областей

Указатели областей предназначены для организации обмена данными, которые находятся в определенных пользователем областях памяти. Указатели областей представляют собой поля параметров, используемых в режиме исполнения системой WinCC flexible для получения информации о расположении и размере областей данных в ПЛК. В процессе обмена данными ПЛК и HMI поочередно обращаются к этим областям данных для выполнения операций чтения и записи. В зависимости от результатов оценки данных, хранящихся в этих областях, контроллер и устройство HMI инициируют выполнение определенных действий.

Указатели областей, используемых в WinCC flexible:

- задание ПЛК
- идентификатор (ID) проекта
- номер экрана
- запись данных
- дата/время
- дата/время ПЛК
- координация

Наличие различных указателей областей определяется используемым устройством HMI.

### Обмен данными между WinCC flexible и системами автоматизации (контроллерами)

Обмен данными с технологическим процессом в WinCC flexible происходит с помощью тегов и указателей областей. Для получения данных технологического процесса устройство HMI отправляет системе автоматизации телеграммы запроса с помощью драйвера связи. Система автоматизации (СА) возвращает затребованные данные устройству HMI в телеграмме ответа.

#### Драйверы связи

Драйвер связи это компонент программного обеспечения, который устанавливает соединение между системой автоматизации и устройством HMI, и таким образом предоставляет возможность передачи значений процесса тегам WinCC flexible. WinCC flexible поддерживает возможность установления соединения различных систем автоматизации с различными драйверами связи.

Пользователь может выбрать интерфейс, профиль и скорость передачи для каждого драйвера связи.

#### Обмен данными между HMI

Для обмена данными между HMI используется протокол SIMATIC HMI HTTP. Этот протокол является компонентом опции "Sm@rtAccess". Протокол можно использовать на ПК с установленной системой исполнения WinCC flexible 2005 и на панелях серии 270. Более подробную информацию можно найти в документации по протоколу SIMATIC HMI HTTP.

#### Обмен данными с использованием универсального и независимого от производителя интерфейса

WinCC flexible предоставляет универсальный и независимый от производителя интерфейс программного обеспечения OPC (OLE для управления процессом). Этот интерфейс делает возможным стандартизованный обмен данными между промышленными и офисными приложениями и технологическим процессом. Более подробную информацию можно найти в документации по OPC.

## 7.2 Элементы пользовательского интерфейса и основная настройка

### 7.2.1 Редактор соединений

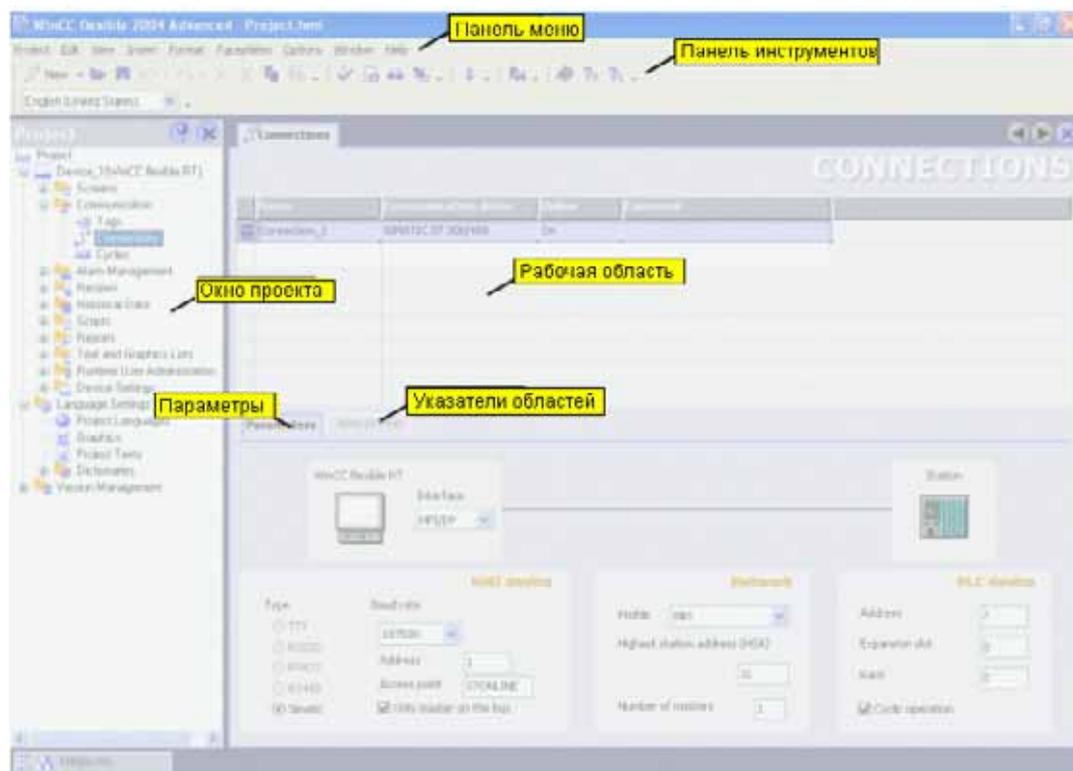
#### Введение

В редакторе соединений создаются соединения и определяются их параметры.

## Открытие редактора

Выберите в окне проекта элемент "Connections [Соединения]" и откройте контекстное меню. Выберите в контекстном меню "New connection [Новое соединения]". В рабочей области будет создано и открыто новое соединение.

## Структура



## Панель меню

Панель меню содержит все команды, необходимые для работы WinCC flexible. Клавиши быстрого вызова команд показаны рядом с командами меню.

## Панели инструментов

Панели инструментов содержат наиболее часто используемые кнопки. Для того чтобы показать или спрятать ту или иную панель инструментов выберите команды "View > Toolbars [Вид > Панели инструментов]". Кнопка панели инструментов используется для того чтобы отобразить или скрыть определенные кнопки этой панели инструментов.

## Рабочая область

Все соединения отображаются в табличном виде в рабочей области. В ячейках таблицы выделяются драйверы связи и редактируются свойства соединений. Для сортировки таблицы по элементам столбца щелкните на заголовке столбца.

## Закладка "Parameters [Параметры]"

На закладке "Parameters [Параметры]" в рабочей области определяются параметры выделенного в таблице драйвера связи. Задайте параметры для HMI, сети и для контроллера.

## Закладка "Area pointer [Указатели областей]"

На закладке "Area pointer [Указатели областей]" в рабочей области определяются указатели областей для соответствующих соединений.

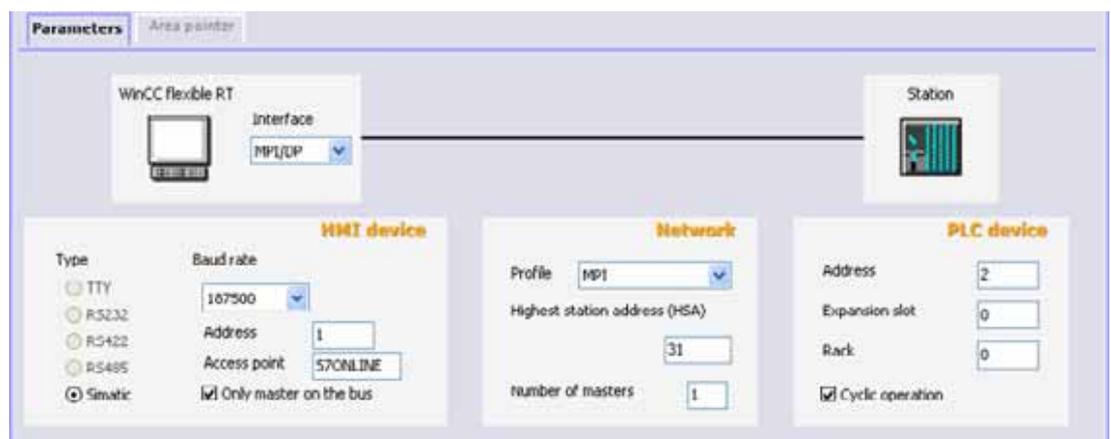
## 7.2.2 Параметры соединений

### Введение

Для определения свойств соединения между устройством HMI и партнером по связи выберите закладку "Parameters [Параметры]" редактора "Connections [Соединения]".

### Структура

Партнеры по связи или коммуникационные партнеры схематично отображаются на закладке "Parameters [Параметры]". На этой закладке представлены области "HMI device [Устройство HMI]", "Network [Сеть]" и "Controller [Контроллер]", в которых вы можете указать параметры используемого интерфейса связи.



Система автоматически устанавливает параметры по умолчанию. При редактировании параметров необходимо следить за их соответствием в пределах всей сети. Более подробную информацию о параметрах можно найти в описании соответствующего протокола.

### 7.2.3 Указатели областей для соединений

#### Введение

На закладке "Area pointer [Указатели областей]" редактора соединений определяются используемые указатели областей.

#### Структура

На закладке "Area pointer [Указатели областей]" представлены две таблицы указателей областей. В таблице "For all connections [Данные для всех соединений]" содержатся те указатели областей, которые создаются в проекте только однажды и могут использоваться только для одного соединения.

В таблице "For each connection [Данные для каждого отдельного соединения]" содержатся указатели областей, которые определяются отдельно для каждого имеющегося соединения.

Parameters: Area pointer							
For all connections:							
Connection	Name	Address	Length	Trigger mode	Acquisition cycle	Comment	
Connection_1	Date/time PLC	DB 1 DBW 0	6	Cyclic continuous	1 min		
<undefined>	Project ID		1	Cyclic continuous	<undefined>		
<undefined>	Screen number		5	Cyclic continuous	<undefined>		
For each connection:							
Active	Name	Address	Length	Trigger mode	Acquisition cycle	Comment	
On	Coordination	DB 1 DBW 12	2	Cyclic continuous	<undefined>		
Off	Data mailbox		5	Cyclic continuous	<undefined>		
Off	Date/time		6	Cyclic continuous	<undefined>		
Off	Job mailbox		4	Cyclic continuous	<undefined>		

Возможные указатели областей определяются используемым устройством HMI. Более подробную информацию об указателях областей можно найти в описании соответствующего протокола.

## Структура системы управления рецептами

### 8.1 Основные сведения о работе с рецептами

#### 8.1.1 Основные сведения о рецептах

##### Введение

Рецепты представляют собой наборы связанных между собой данных, например, данные о сырье для производства продукта. Вы можете передать этот набор данных с устройства HMI в контроллер одним действием, например, для того, чтобы изменить вариант производимого продукта. Если данные для производства были введены путем непосредственного программирования оборудования, то можно, например, передать их на устройство HMI и записать в рецепт.

##### Основные сведения

Рецепты создаются в редакторе рецептов ("Recipes"). Для отображения и редактирования рецептов на устройстве HMI необходимо создать на экране процесса либо окно рецептов, либо экран рецептов.

Существуют следующие возможности ввода данных:

- Ввод данных в режиме исполнения проекта  
Если необходимо часто изменять или корректировать производственные данные, вы можете при исполнении проекта ввести данные непосредственно на устройстве HMI или на самом оборудовании. Примером является режим динамической настройки оборудования ("Teach in"). Перемещаемые компоненты оборудования можно перевести в требуемое положение непосредственно на оборудовании. Затем эти данные передаются с ПЛК на устройство HMI и записываются в рецепт.
- Импорт данных в режиме исполнения проекта  
Если производственные данные хранятся, например, в базе данных на сервере вы можете в режиме исполнения импортировать их на устройство HMI в виде CSV-файла.
- Ввод данных при разработке проекта  
Если производственные данные уже имеются в распоряжении на этапе разработки проекта или являются фиксированными данными, их можно ввести или импортировать при создании проекта с помощью редактора рецептов.

## Примеры использования рецептов

Рецепты используются в производственных отраслях промышленности и в машиностроении. Следующие два примера иллюстрируют типичное применение рецептов с использованием всех функциональных возможностей системы разработки WinCC flexible, связанных с рецептами:

- **Настройка параметров технологического оборудования**

Одна из областей использования рецептов это настройка параметров оборудования на производстве. На оборудовании изготавливаются (выпиливаются) деревянные доски различной величины по указанным размерам, и в них затем просверливаются отверстия. В зависимости от размера одной доски направляющие и сверла должны перемещаться на другую позицию. Данные о положении направляющих и сверл хранятся в виде записей данных в рецепте. Если необходимо обработать доску другого размера, вы можете задать параметры оборудования (данные о положении) в режиме динамической настройки ("Teach in"). Данные о новом положении направляющих и сверл передаются непосредственно с ПЛК на устройство HMI и сохраняются в виде записи данных.

- **Серийное производство**

Еще один вариант применения рецептов – в пищевой промышленности при изготовлении продуктов партиями. На станции смешивания установки по производству фруктовых соков производятся сок, нектар и фруктовые напитки с различными вкусами. Ингредиенты всегда одинаковые, отличаются только пропорции при их смешивании. Каждому напитку соответствует свой рецепт. Описание количества каждого смешиваемого продукта соответствует записи данных. Все данные, требуемые для смешивания ингредиентов, могут быть переданы системе управления путем нажатия одной кнопки.

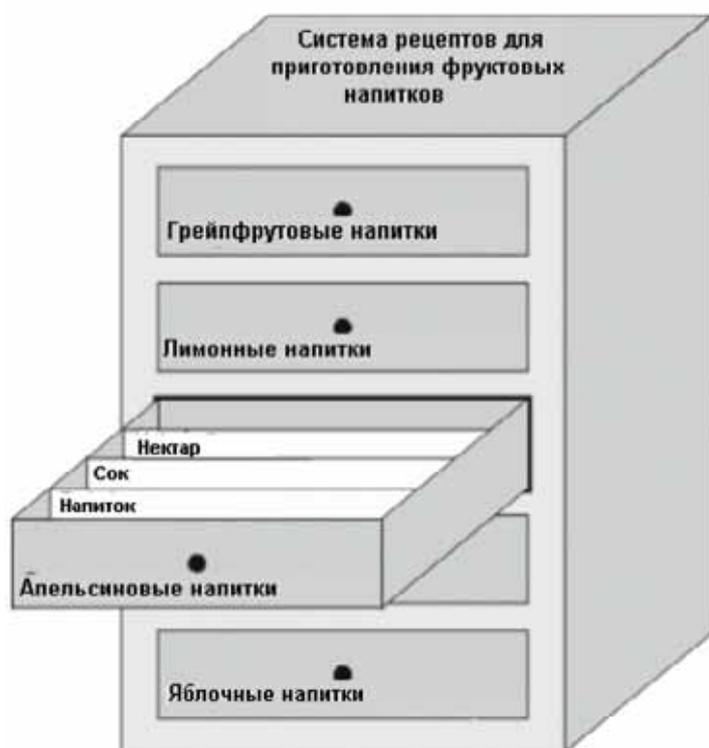
## 8.1.2 Структура рецепта

### Введение

Часто существует несколько вариантов рецептов. Например, варианты продукта могут отличаться объемом или составом продукта. Это условие отражено в рецепте.

### Основные сведения

Рецепт состоит из записей данных, содержащих значения процесса. Структура рецепта показана с использованием модели картотеки или шкафа для хранения документов.



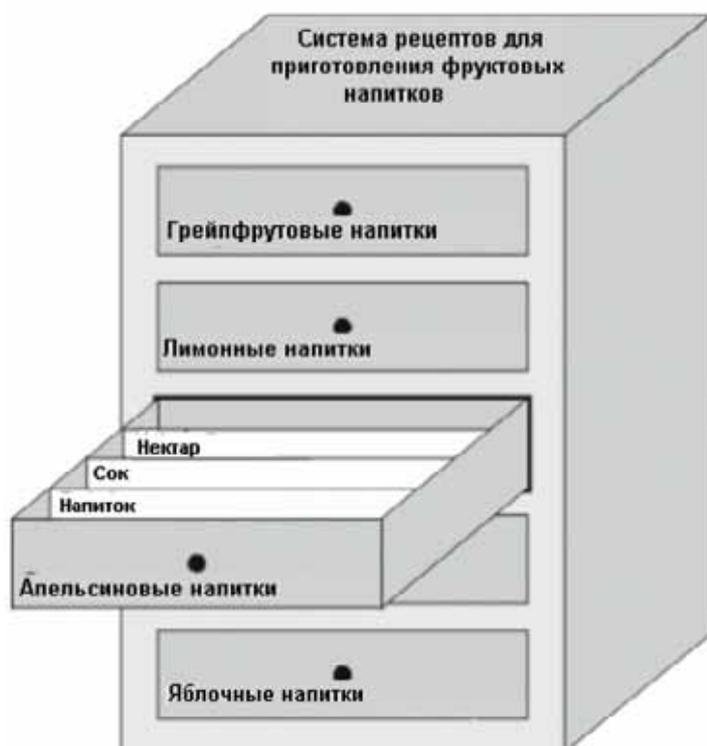
Каждый рецепт представляет отдельный ящик шкафа-картотеки и описывает один продукт. Если на установке по приготовлению фруктовых напитков производятся напитки с апельсиновым, яблочным вкусом и вкусом тропических фруктов, вам необходимо создать рецепт для каждого из этих напитков.

Рецепт состоит из элементов рецепта. Элемент рецепта содержит отображаемое имя и тег. Имена показаны в записях данных рецепта и отображаются на устройстве HMI в окне рецептов. Соответствующие значения тегов в режиме исполнения считываются из контроллера или передаются в контроллер.

### 8.1.3 Структура записи данных рецепта

#### Введение

Запись данных соответствует в рассматриваемой модели регистрационной карточке или папке в отдельном ящике шкафа-картотеки, а реально - одному варианту продукта. Если на установке по приготовлению фруктовых напитков производятся сок, нектар и фруктовые напитки, то для каждого варианта продукта создается запись данных рецепта. Варианты продукта отличаются различными пропорциями смешивания ингредиентов.



Запись данных рецепта содержит набор значений тегов, определенных в рецепте. Значения вводятся в поля ввода. Ввод значений может выполняться на этапе проектирования или во время исполнения проекта.

Elements		Data records					
	Name	Display name	Numb... ▲	Water	Concentrate	Sugar	Aroma
☰	Beverage	Beverage	1	30	70	45	600
☰	Nektar	Nektar	2	50	50	10	300
☰	Juice	Juice	3	5	95	3	100

Для производства продукта соответствующая запись данных рецепта передается с устройства HMI в подключенный к нему контроллер. Значения записи данных рецепта не могут быть изменены на устройстве HMI, если инженер разработчик не обеспечил такую возможность при создании проекта.

## Редактирование записей данных рецепта

Записи данных рецепта могут редактироваться на устройстве HMI в процессе создания проекта или в режиме его исполнения.

- Рецепты создаются в процессе проектирования на закладке "Elements [Элементы]" редактора рецептов. Значения тегов в записях рецептов вводятся на закладке "Data records [Записи данных]".
- В режиме исполнения значения записей данных рецепта могут вводиться непосредственно на устройстве HMI, либо импортироваться с использованием CSV-файла. Данные записей могут быть также экспортированы в файл CSV.

### 8.1.4 Настройка функционирования рецептов

#### Введение

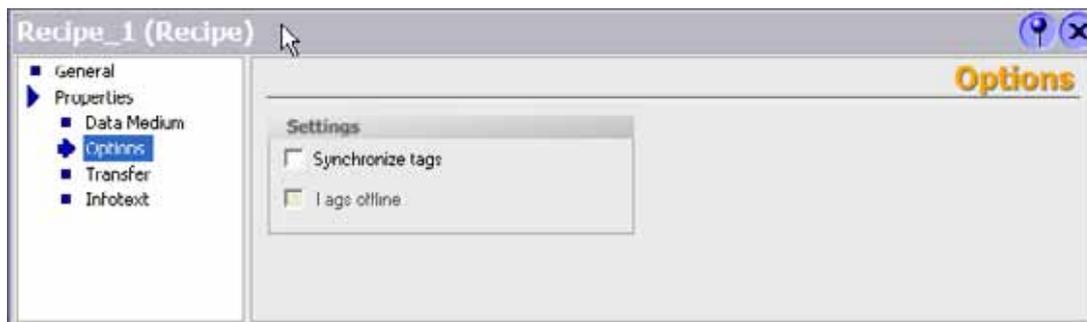
Параметры функционирования рецептов определяются в соответствии с предполагаемым их использованием. Параметры настройки, при которой записи значений в записи данных рецепта будет выполняться на устройстве HMI без вмешательства в текущий процесс, будут отличаться от параметров настройки, при которой значения записей рецепта будут устанавливаться непосредственно на оборудовании.

#### Основные сведения

Параметры настройки функционирования рецепта определяют вариант функционирования тегов, используемых в рецепте. На приведенном ниже рисунке показаны основные отличия при работе с записями данных рецепта.



Параметры определяются в области "Settings [Настройка]" окна свойств:



### Вариант настройки 1: рецепт без включения опции "синхронизация тегов"

Данные считанной записи рецепта отображаются и могут редактироваться только в окне рецепта. Использование этих же самых тегов вне рецепта не оказывает влияния на их значения в рецепте.

### Вариант настройки 2: рецепт с включенными опциями "синхронизация тегов" и "теги в режиме offline"

При включении опции "Synchronize tags [Синхронизация тегов]" данные записи, считанной из контроллера или другого носителя информации, будут записаны или считаны из тегов, используемых в рецепте.

При включении опции "Offline" введенные данные записываются в теги, но не передаются непосредственно в контроллер.

### Вариант настройки 3: рецепт с включенной опцией "синхронизация тегов" и выключенной опцией "теги в режиме offline"

При включении опции "Synchronize tags [Синхронизация тегов]" данные записи, считанной из контроллера или другого носителя информации, будут записаны или считаны из тегов, используемых в рецепте.

Введенные или считанные данные немедленно передаются в контроллер:

### Синхронизация с контроллером

При синхронной передаче и в контроллере и в устройстве HMI устанавливается бит состояния в общей области данных. Этот механизм может использоваться для предотвращения нежелательной перезаписи данных в любом направлении в вашей программе управления. Диапазон адресов области данных определяется для каждого контроллера по отдельности на закладке "Range pointer [Указатели областей]" в редакторе "Connections [Соединения]".

Особенности использования синхронной передачи записей данных:

- Контроллер является "активным партнером" при передаче записей данных рецепта.
- В контроллере анализируются данные, содержащие номер и имя рецепта, а также номер и имя записи рецепта.
- Передача данных инициируется с помощью системной функции или задания ПЛК, например, с помощью системной функции "SetDataRecordToPLC" и "GetDataRecordFromPLC", или с помощью заданий ПЛК "Set\_Data\_Record\_To\_PLC" и "Get\_Data\_Record\_From\_PLC."

Для синхронизации передачи записей данных между устройством HMI и контроллером, необходимо выполнение следующих условий:

- Указатель диапазона "Data mailbox [Блок данных задания ПЛК]" отображается в группе элементов "Range pointers [Указатели диапазонов]" в окне проекта.
- Контроллер, с которым синхронизирована передача данных, указан в свойствах рецепта.

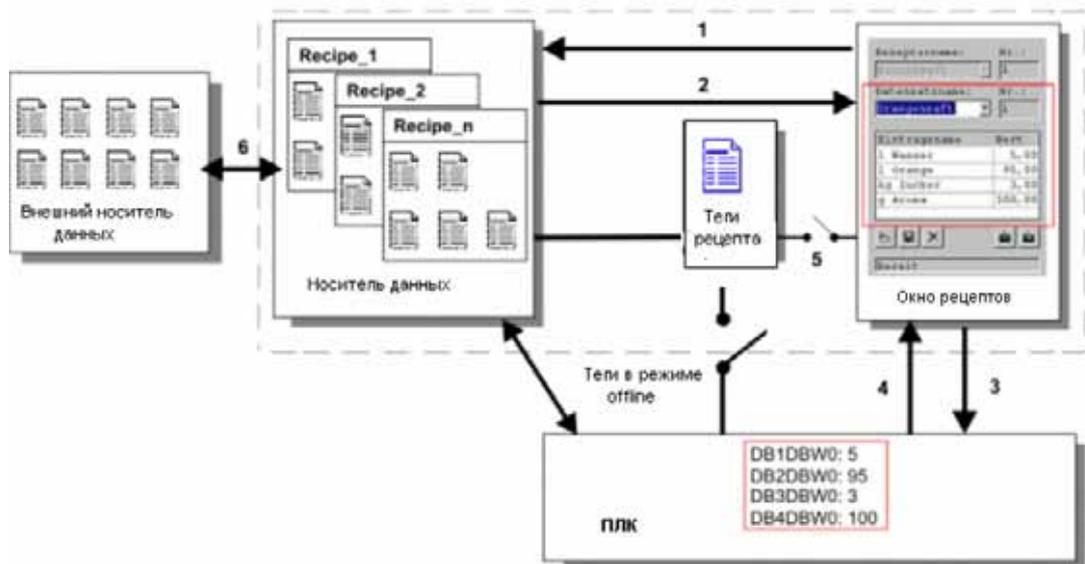
### 8.1.5 Передача записей данных рецепта

#### Введение

Записи данных рецепта могут передаваться в режиме исполнения проекта между внешними носителями данных, например, флэш-памятью, устройством HMI и контроллером.

#### Основные сведения

На рисунке, приведенном ниже, показано, каким образом могут быть переданы записи данных. В окне проекта определяются соответствующие функциональные возможности для передачи записей данных. На экране рецепта могут использоваться предназначенные для этого системные функции.



На устройстве HMI записи данных рецепта хранятся на носителе данных, таком как флэш-память или жесткий диск. Записи данных рецепта редактируются в окне рецептов или на экране рецептов, отображаемом на устройстве HMI.

(1) Сохранение: Путем выполнения функции "Save [Сохранение]" значения, измененные в окне рецептов или на экране рецептов, заносятся в запись данных.

(2) Загрузка: Функция "Load [Загрузка]" используется для обновления значений тегов рецепта, отображаемых на экране рецептов значениями из записи данных рецепта, хранящихся на носителе данных. В результате выполнения этой функции изменяются значения, отображаемые на экране рецептов. При повторном выделении записи данных в окне рецептов выполняется функция загрузки ("Load").

(3) Запись в контроллер: Изменения значений в окне рецептов и на экране рецептов загружаются в ПЛК путем вызова функции "Write to controller [Запись в контроллер]".

(4) Чтение из контроллера: Выполнение функции "Read from controller [Чтение из контроллера]" позволяет обновить отображаемые в окне рецептов или на экране рецептов значения значениями из контроллера. В результате выполнения этой функции изменяются значения, отображаемые в окне или на экране рецептов.

(5) Синхронизация с контроллером: С помощью этой функции вы можете синхронизировать значения, отображаемые в окне рецептов со значениями тегов рецепта. После такой синхронизации и теги рецепта и окно рецептов содержат текущие обновленные значения. При отключении опции "Variables offline [Теги в режиме offline]" текущие значения также передаются в контроллер.

(6) Импорт, экспорт: Запись данных может быть экспортирована на внешний носитель данных для последующей обработки, например, в MS Excel. Запись данных хранится в формате \*.csv.

## 8.2 Элементы пользовательского интерфейса и основная настройка

### 8.2.1 Редактор рецептов

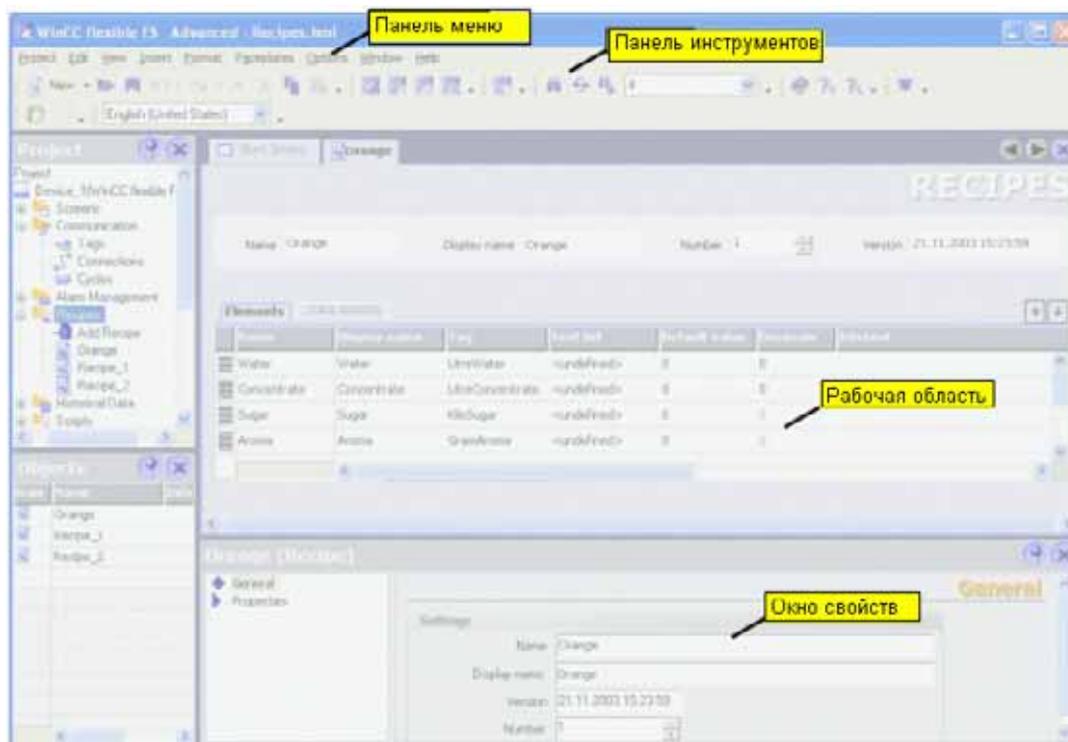
#### Введение

В редакторе рецептов создаются и редактируются рецепты данных. Кроме того, редактор рецептов можно использовать для ввода значений в уже существующие записи данных рецепта.

#### Открытие редактора рецептов

Откройте редактор рецептов либо путем создания нового рецепта, либо открыв существующий рецепт.

## Структура редактора рецептов



### Панель меню

Панель меню содержит все команды, необходимые для работы WinCC flexible. Возможные комбинации клавиш для быстрого вызова команд меню показаны рядом с соответствующими командами.

### Панель инструментов

На панели инструментов представлены наиболее важные команды меню.

### Рабочая область

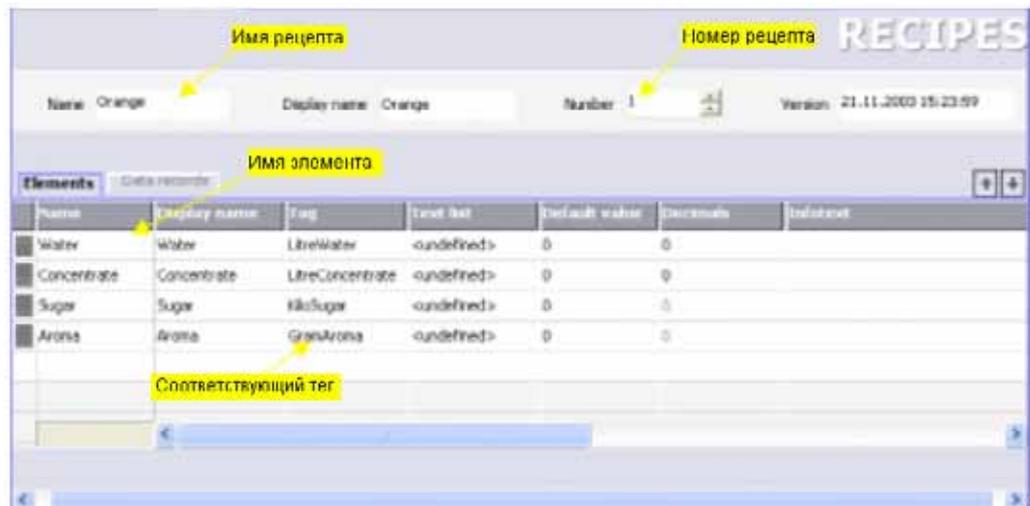
В этой области создаются и редактируются элементы рецептов и записи данных рецептов. Рецепт определяется на закладке "Elements [Элементы]".

### Окно свойств

В этом окне определяются свойства рецепта. Дополнительную информацию об определении свойств рецептов можно найти в разделе "Recipe settings [Определение свойств рецептов]."

## 8.2.2 Элементы рецепта

### Структура редактора на закладке Elements [Элементы]



### Элементы рецепта

Отдельные элементы рецепта кратко описаны ниже.

#### Имя рецепта

Имя рецепта однозначно определяет рецепт в пределах проекта.

#### Отображаемое имя

Отображаемое имя появляется при исполнении проекта в окне рецептов и может быть выведено на одном из нескольких языков. Можно использовать описательные имена и обозначения, связанные с конкретным продуктом, например, "FruitJuice\_Orange."

#### Номер рецепта

Номер рецепта однозначно определяет рецепт в пределах проекта.

#### Версия

Версия определяет дату и время внесения самого последнего изменения в рецепт.

#### Имя элемента

Имя элемента однозначно определяет элемент рецепта в пределах этого рецепта. Можно использовать описательные имена или обозначения, например, название оси для обозначения направляющей или имя "вкус" для обозначения ингредиента.

### Назначенный тег

Каждому элементу рецепта назначается или ставится в соответствие тег рецепта, в который при исполнении проекта записывается значение записи данных рецепта.

### Значение по умолчанию

Значение по умолчанию – это значение, которое вводится при создании новой записи данных рецепта.

### Текстовые списки

Значением элемента текстового списка является текст. Этот текст затем может отображаться в поле вывода.

### Десятичные разряды

Это число обозначает количество десятичных разрядов при отображении значения записи данных рецепта в режиме исполнения.

### Текст подсказки

Здесь вы можете ввести текст подсказки об элементе рецепта. Пользователь может воспользоваться подсказкой в режиме исполнения.

## 8.2.3 Записи данных рецепта

### Структура редактора на закладке Data records [Записи данных]

Name	Display name	Number	What	Concentrate	Sugar	Amount
Beverage	Beverage	1	30	70	45	600
Milkshake	Milkshake	2	50	50	10	300
Juice	Juice	3	5	95	3	100

### Элементы записи данных рецепта

На закладке "Data records [Записи данных]" отображаются следующие элементы:

### **Имя записи данных рецепта**

Имя записи данных рецепта однозначно определяет запись данных в пределах рецепта.

### **Отображаемое имя**

Отображаемое имя записи данных рецепта появляется при исполнении проекта в окне рецептов и может быть выведено на одном из нескольких языков. Можно использовать описательные имена и обозначения, связанные с конкретным продуктом, например, название продукта

### **Номер записи данных рецепта**

Номер записи данных рецепта однозначно определяет запись данных в пределах рецепта.

### **Введенные значения**

При исполнении проекта в запись данных рецепта вводятся значения. При передаче проекта на устройство HMI также передаются записи данных рецепта. Если на устройстве HMI уже хранятся записи данных, в зависимости от параметров настройки передачи, после вывода подсказки пользователю новые значения записываются на место старых.

### **Текст подсказки**

Здесь вы можете ввести текст подсказки об элементе рецепта. Пользователь может воспользоваться подсказкой в режиме исполнения.

## **8.2.4 Определение свойств рецептов**

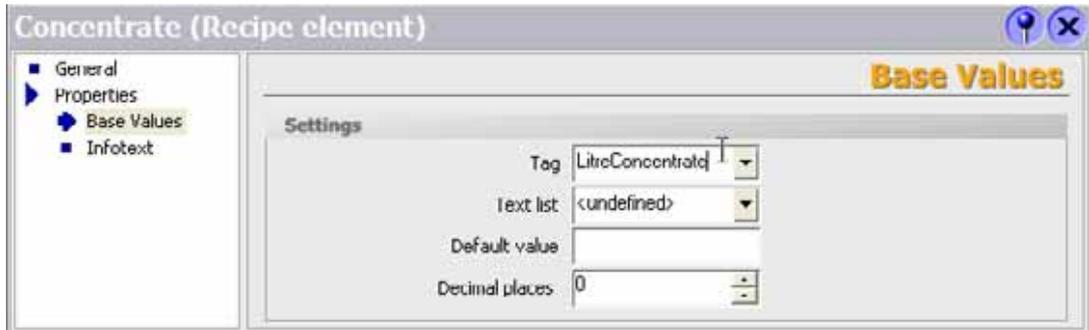
### **Введение**

В окне свойств определяются свойства рецепта.

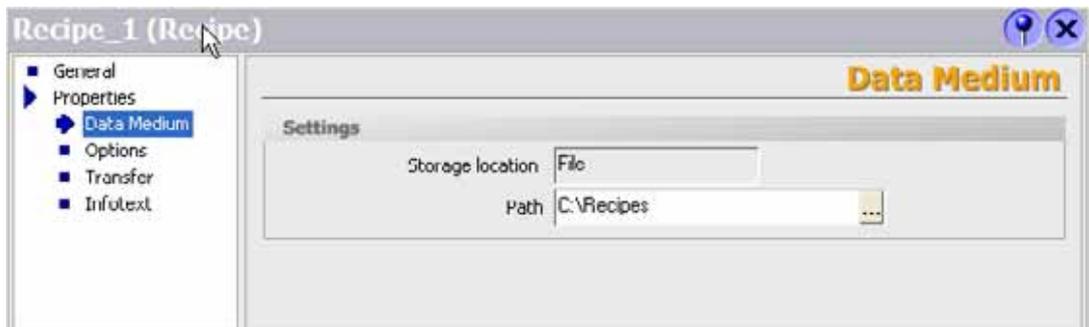
### **Основные сведения**

Содержимое окна свойств зависит от параметров настройки, выбранных в редакторе рецептов. При редактировании элементов рецепта или записей данных рецепта на закладках "Elements [Элементы]" или "Data records [Записи данных]" вы также можете изменить содержимое окна свойств, отображаемое на экране.

Для изменения свойств рецепта щелкните на поле "Recipe name [Имя рецепта]" или "Recipe number [Номер рецепта]" в редакторе рецептов. В окне свойств можно выполнять следующие действия:



Например, в группе свойств "Data medium [Носитель данных]" определяется, где будет храниться файл, содержащий записи данных рецепта. Возможности выбора зависят от используемой операторской панели. В зависимости от конкретного устройства операторской панели можно выбрать флэш-память или карту памяти MMC операторской панели. При работе в среде исполнения WinCC flexible сохраните файл на жестком диске компьютера. Введите путь к файлу непосредственно или выберите требуемый носитель данных, используя диалоговое окно.



Функционирование рецепта в режиме исполнения определяется путем выбора соответствующих опций в группах свойств. Например, в группе свойств "Transfer [Передача]" в области "Settings [Настройка]" можно определить, будут ли значения тегов рецепта в режиме исполнения проекта немедленно передаваться в ПЛК.



## 8.3 Просмотр и редактирование рецептов в режиме исполнения

### 8.3.1 Просмотр и редактирование рецептов в режиме исполнения

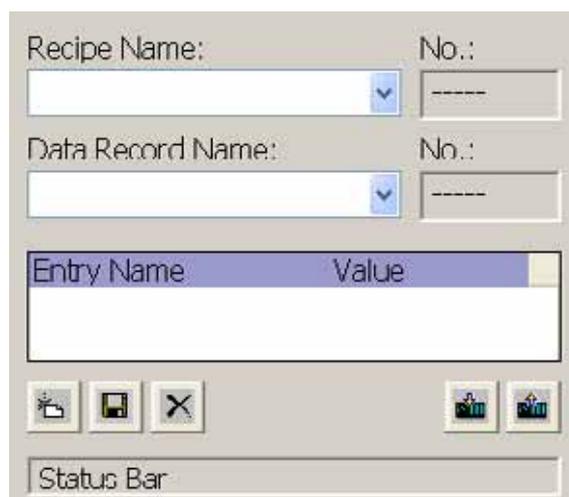
#### Введение

Система разработки WinCC flexible предоставляет возможность создания двух объектов для просмотра и редактирования рецептов и соответствующих записей данных на устройстве HMI в режиме исполнения:

- Окно рецептов
- Экран рецепта

#### Окно рецептов

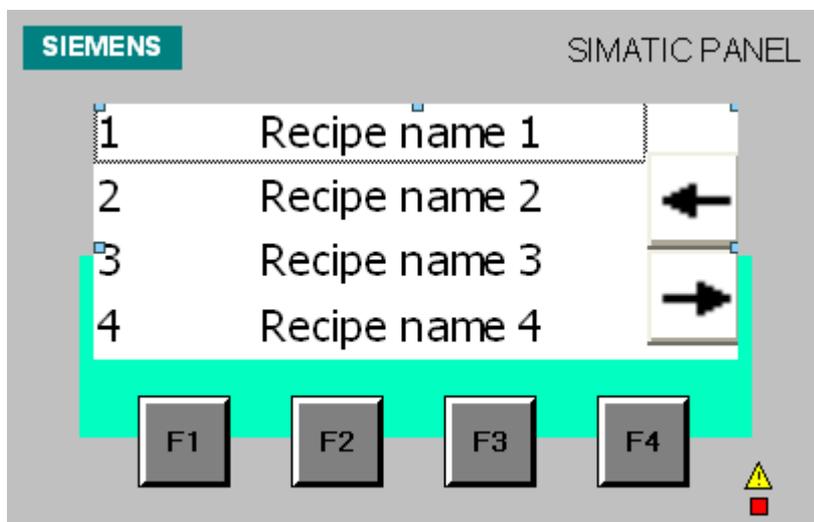
Окно рецептов это экранный объект, конфигурируемый в редакторе экранов. Вы можете, например, определить функции, которые можно выполнить в окне рецептов режиме исполнения:



Записи рецепта представлены в окне рецептов в табличной форме. Окно рецептов особенно удобно, если записи данных невелики по размеру или необходимо изменить только несколько значений.

### Простое окно рецептов

На устройствах HMI с размером экрана дисплея менее 6" (например, OP 77B), для отображения и редактирования рецептов используется простое окно рецептов.



Простое окно рецептов состоит из трех областей:

- Выбор рецепта
- Выбор записи данных рецепта
- Элементы рецепта

В простом окне рецептов каждая из областей показана отдельно. Просмотр рецептов всегда начинается с выбора рецепта.

### Экран рецепта

Экран рецепта представляет собой экран процесса, на котором путем конфигурирования и настройки полей ввода/вывода и других экранных объектов в редакторе экранов создается пользовательская экранная форма для ввода значений. Это позволяет вводить данные в контексте управления технологическим процессом. Поля ввода/вывода для рецепта могут быть распределены между несколькими экранами рецепта, что позволяет вам организовать элементы рецепта по темам. Функции для работы с экранами рецептов должны быть явно определены на экранах процесса.

Litre Water	50	Recipe name	No.:
Litre Concentrate	50	Orange	1
Kilo Sugar	30	Data record name	No.:
Gram Aroma	50	Nectar	2

Save	Data to PLS
Load	Data from PLS

### 8.3.2 Основные сведения об окне рецептов

#### Введение

Окно рецептов представляет собой экранный объект, используемый для отображения и редактирования записей данных рецепта в режиме исполнения.

Recipe Name:	No.:					
<input type="text"/>	-----					
Data Record Name:	No.:					
<input type="text"/>	-----					
<table border="1"><thead><tr><th>Entry Name</th><th>Value</th></tr></thead><tbody><tr><td> </td><td> </td></tr></tbody></table>		Entry Name	Value			
Entry Name	Value					
<table border="1"><tr><td></td><td></td><td></td><td></td><td></td></tr></table>						
Status Bar						

#### Настройка

Функции окна рецептов определяются пользователем.

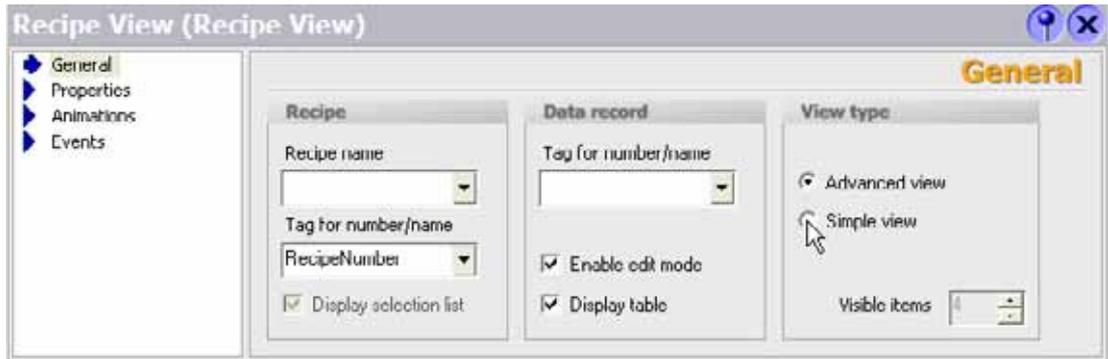
Вы можете определить, будет ли окно рецептов служить только для выбора рецепта или в нем можно будет также редактировать рецепт.

### 8.3.3 Основные сведения о простом окне рецептов

#### Введение

На устройствах HMI с размером экрана дисплея менее 6" (например, OP 77B), для отображения и редактирования рецептов используется простое окно рецептов.

Такое окно рецептов можно использовать на всех других устройствах HMI.

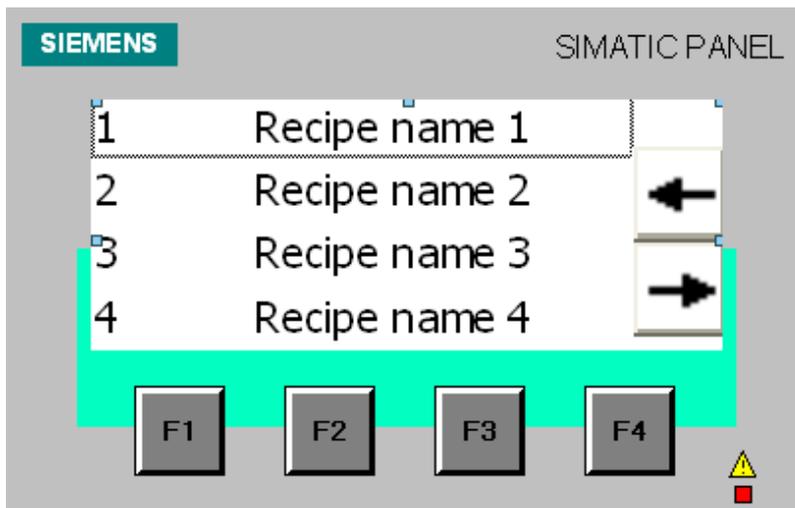


#### Структура и функции окна

Простое окно рецептов состоит из трех областей:

- Выбор рецепта
- Выбор записи данных рецепта
- Элементы рецепта

В простом окне рецептов каждая из областей показана отдельно. Просмотр рецептов всегда начинается с выбора рецепта.



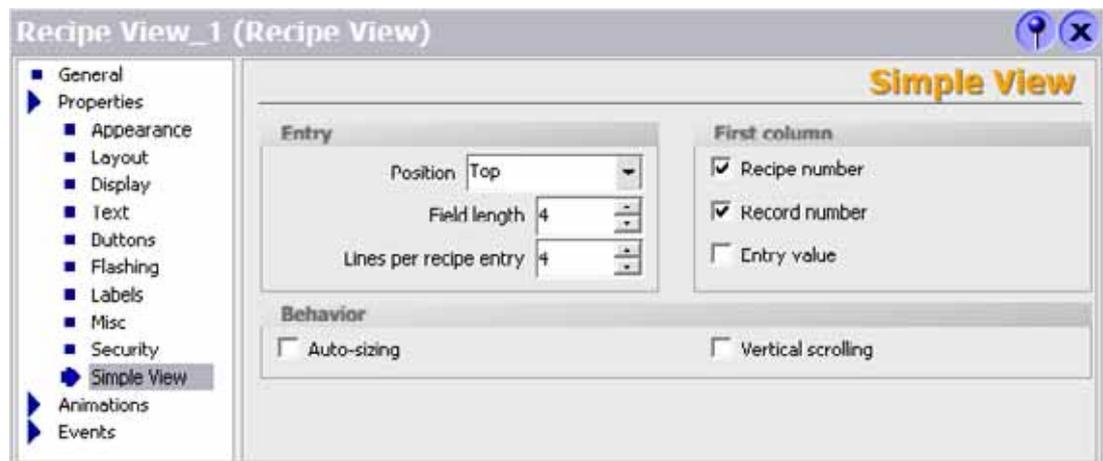
Выбор команды в каждой из отображаемых областей может быть сделан с помощью кнопки ↵. В списке выбираемых команд перечислены все команд, которые можно использовать в отображаемой в данный момент области. Каждой команде ставится в

соответствие номер, который вы можете использовать для вызова команды непосредственно из списка (без клавиши <Enter>).

## Настройка

Для настройки простого окна выберите "Recipe view [Окно рецептов]" в группе "Enhanced objects [Расширенный набор объектов]" окна инструментов.

Элемент "Simple view [Простое окно]" можно также выбрать в окне свойств при определении свойств окна рецептов.



Для работы оператора в окне рецептов можно использовать функциональные клавиши HMI для следующих системных функций:

- RecipeViewMenu: Открывает окно выбора команд меню
- RecipeViewOpen: В зависимости от выбора отображаются записи рецепта или элементы рецепта
- RecipeViewBack: Возврат к предыдущему отображению

### 8.3.4 Элементы операторского управления в окне рецептов

#### Элементы операторского управления окна рецептов

В окне рецептов могут быть настроены следующие элементы операторского управления:

Элемент операторского управления	Функция
	Выводит текст подсказки оператору для данного окна рецептов.
	Создает новую запись рецепта, которая отображается в указанном окне рецептов. Переменные записи рецепта принимают значения, которые были определены при создании рецепта как "Basic value [Значения по умолчанию]".
	Сохраняет запись данных рецепта, которая в данный момент отображается в окне рецептов. Область памяти, в которой будет сохранена запись данных, определяется в окне свойств в группе "Properties [Свойства]" > "Data medium [Носитель данных]".
	Сохраняет запись данных, отображаемую в данный момент в окне рецептов под новым именем. Область памяти, в которой будет сохранена запись данных, определяется в окне свойств в группе "Properties [Свойства]" > "Data medium [Носитель данных]".
	Удаляет запись данных, которая отображается в окне рецептов, с носителя данных устройства HMI.
	Синхронизирует запись рецепта, отображаемую в данный момент в окне рецептов, путем обновления значений соответствующих тегов. В процессе синхронизации только те значения, которые были изменены в окне рецептов, записываются в соответствующие теги. Затем значения считываются из тегов и используются для обновления окна рецептов.
	Передает запись рецепта, отображаемую в данный момент в окне рецептов, в контроллер.
	Передает запись рецепта, загруженную в ПЛК, на устройство HMI и отображает ее в окне рецептов.

Кнопкам управления можно поставить в соответствие системные функции. Это можно сделать, например, в том случае, когда кнопки в окне рецептов не будут использоваться или устройство HMI не предоставляет возможности сенсорного управления.

#### Замечание

Элементы операторского управления показаны в виде функций меню в простом окне рецептов.

### 8.3.5 Функционирование окна рецептов в режиме исполнения

#### Смена экрана

Если вы хотите перейти к другому экрану, но не сохранили изменения в данных рецепта, отображаемых в окне рецептов, появится соответствующая подсказка. При этом для идентификации данных, которые еще не были сохранены, будут указаны имя рецепта и имя записи рецепта, содержащей эти не сохраненные данные.

Если вы переходите к экрану, который содержит окно рецептов с загруженными данными рецепта, данные рецепта будут обновлены автоматически.

#### Работа в окне рецептов с помощью программируемых клавиш

С содержимым окна рецептов можно работать с помощью функциональных клавиш. Это удобно, например, когда устройство HMI не имеет сенсорных элементов управления. Системные функции позволяют ставить в соответствие клавишам устройства HMI такие функции как "Save data record [Сохранить запись данных]".

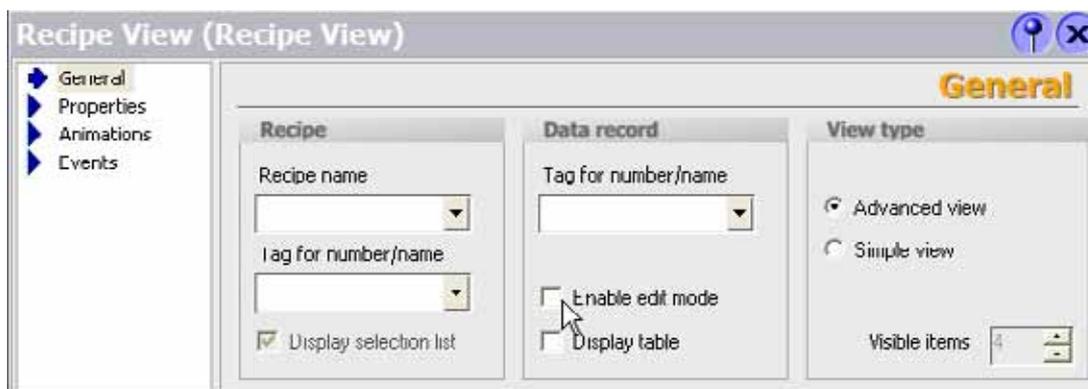
#### Отображение импортированных данных рецепта

Если вы открываете окно рецептов во время выполнения процедуры импорта данных, будут отображаться только данные рецепта, импорт которых уже завершен. Обновление импортируемых данных в окне рецептов не выполняется автоматически. Для того чтобы в окне были отображены все импортированные данные рецепта, следует открывать его только после появления сообщения о том, что импорт данных был успешно выполнен. Альтернативно, можно обновить содержимое окна рецептов после успешного завершения процедуры импорта.

### 8.3.6 Варианты настройки окна рецептов

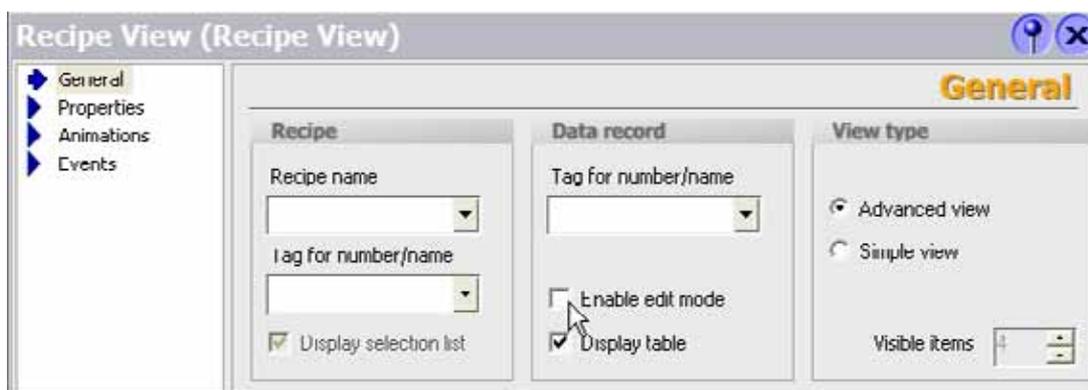
#### Настройка окна рецептов в виде выпадающего списка

Окно рецепта может использоваться на экране процесса в виде выпадающего списка для выбора рецептов или записей данных рецептов (или обоих списков). С этой целью необходимо скрыть все элементы операторского управления и таблицу записей данных рецепта. Тогда на экране процесса будут отображены только два выпадающих списка, из которых можно выбирать рецепты или записи данных рецепта.



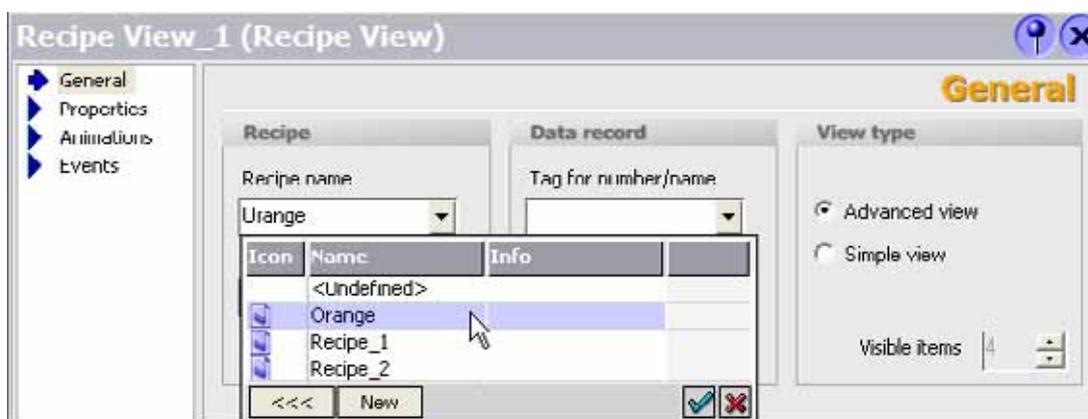
### Отображение значений записей данных с отключенной опцией редактирования

Если вы хотите только просматривать данные рецепта в окне рецептов, вы можете отключить возможность редактирования записей данных рецепта. С этой целью отключите опцию "Enable edit mode [Включить режим редактирования]".



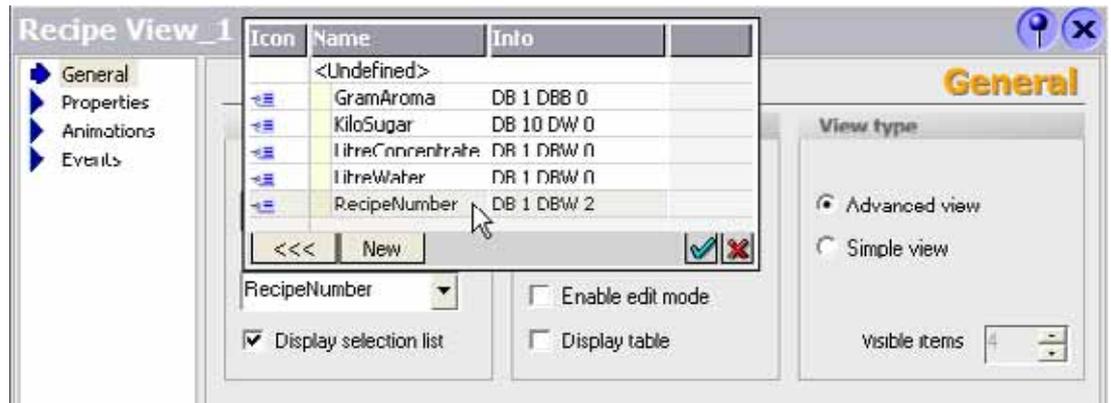
### Отображение конкретного рецепта

Если вы хотите, чтобы на экране процесса был разрешен доступ только к записям данных конкретного рецепта, этот рецепт можно указать в окне рецептов.



### Запись в тег номера или имени рецепта и номера или имени записи данных рецепта

В окне рецептов можно связать рецепт или запись данных рецепта с тегом. При этом при выборе рецепта или записи данных рецепта, соответствующее имя или номер сохраняются в теге. Наоборот, значение тега можно использовать для выбора рецепта или записи данных рецепта путем ввода соответствующего значения. Тип тега определяет, будет ли сохранено имя или номер тега. Если необходимо, чтобы сохранялось имя тега, необходимо определить тег типа STRING [СТРОКА]. Вы можете, например, передать тег системной функции в качестве параметра.



### 8.3.7 Основные сведения об экране рецепта

#### Введение

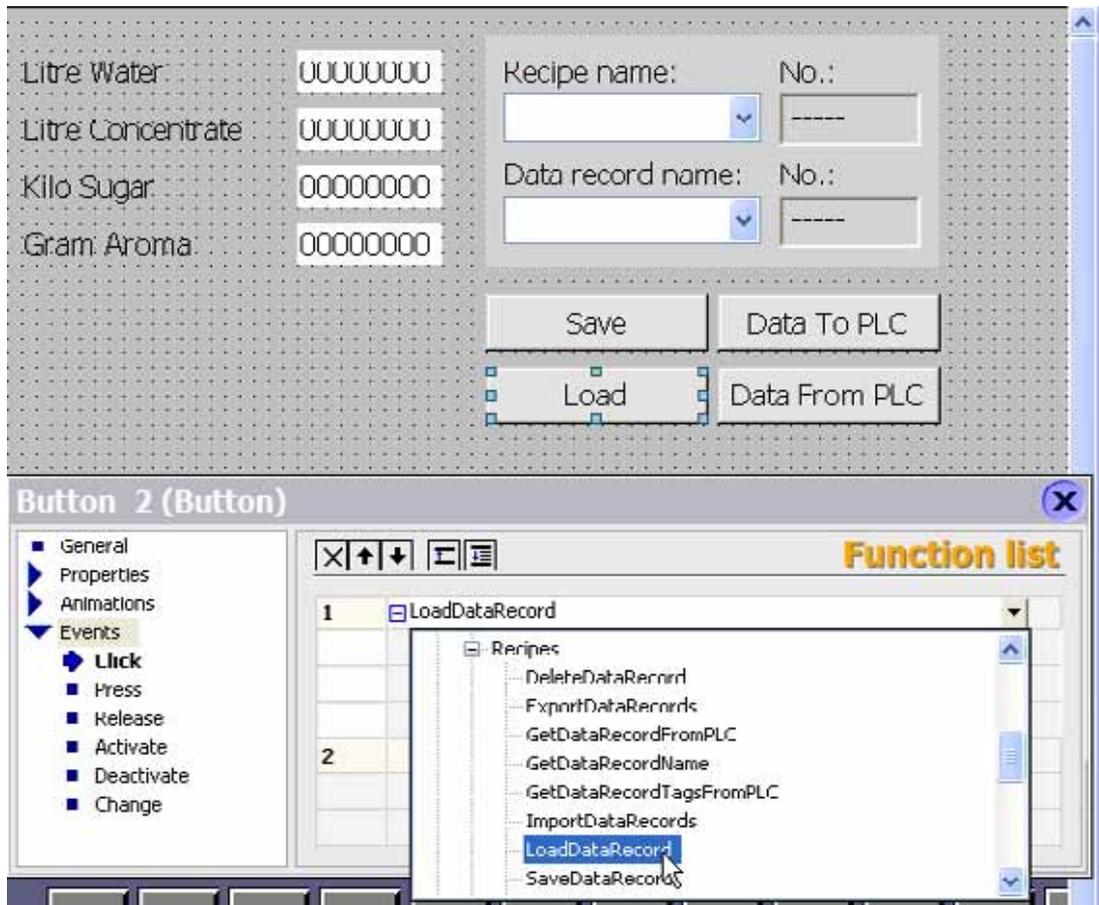
Экран рецепта представляет собой экран процесса, на котором с помощью редактора экранов сконфигурирована пользовательская форма или шаблон для ввода данных. Форма ввода создается из полей ввода/вывода и других экранных объектов. Для настройки функциональных возможностей работы с рецептами таких как, например, сохранение записей данных рецептов, используются системные функции.

---

#### Замечание

Экран рецепта может быть создан для панелей TP 170В и моделей более высокого уровня.

---



### Основные сведения

Возможность произвольного конфигурирования экрана рецепта позволяет создать экран, на котором будут отображаться данные рецепта так, как это удобно пользователю. Например, для отображения данных большого рецепта можно создать несколько экранов рецептов, данные которых будут распределены по темам. Данные отображаются наглядно с помощью графических экранных объектов.

- Распределение данных рецепта между несколькими экранами процесса по темам

Вы можете распределить данные записей рецепта, содержащих большое количество элементов между несколькими экранами рецепта. Например, для каждой секции установки вы можете создать экран процесса, содержащий формы ввода данных соответствующих записей рецепта. Одним из примеров может быть визуализация технологического процесса для производства столешниц. Здесь можно создать отдельные экраны процесса для операций подачи пиломатериалов, обстругивания, распиливания, высверливания, полировки и упаковки.

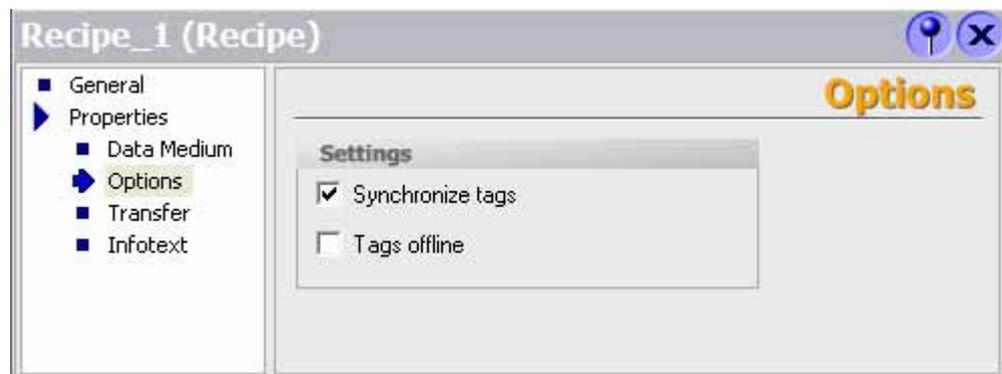
Распределение данных рецептов между несколькими экранами процесса удобно для устройств HMI с маленькими дисплеями. В частности, это позволяет избежать прокрутки таблиц в режиме реального времени.

- Наглядная имитация работы машины

Вы можете наглядно смоделировать работу машины на экране процесса с помощью экранных объектов. Это позволяет отобразить процесс определения параметров более наглядно, помещая поля ввода/вывода параметров управления непосредственно рядом с соответствующими элементами установки, такими как, например, направляющие и режущие элементы. Таким образом, значения процесса непосредственно привязываются к экранной модели установки.

## Параметры настройки

В окне свойств необходимо установить флажок опции "Synchronize tags [Синхронизация тегов]", если необходимо иметь возможность вводить значения записей данных рецепта в поля ввода/вывода, созданные вне окна рецептов.



Если необходимо, чтобы введенные значения в режиме исполнения проекта немедленно передавались в подключенный ПЛК, следует в окне свойств снять флажок опции "Tags offline [Теги в режиме offline]".

Для того чтобы иметь возможность включать и отключать немедленную передачу введенных значений в режиме исполнения, следует использовать системную функцию "SetRecipeTags".

## Системные функции

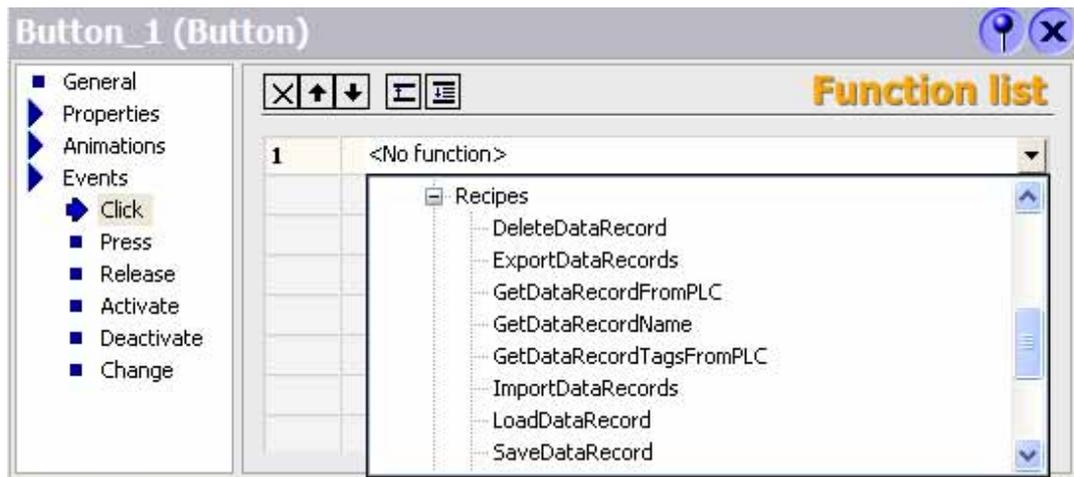
Для операторского контроля и управления с использованием экрана рецепта можно использовать следующие системные функции.

- ImportDataRecords [Импорт записей данных]
- ExportDataRecords [Экспорт записей данных]
- LoadDataRecord [Загрузка записи данных]
- SaveDataRecord [Сохранение записи данных]
- SetDataRecordTagsToPLC [Записать теги записи рецепта в ПЛК]
- GetDataRecordTagsFromPLC [Считать теги записи рецепта из ПЛК]

Для операторского управления с использованием окна рецептов, если таковое было сконфигурировано на экране процесса, можно использовать следующие системные функции.

- RecipeViewSaveDataRecord [Окно рецептов Сохранить запись данных]
- RecipeViewSaveAsDataRecord [Окно рецептов Сохранить как запись данных]
- RecipeViewSynchronizeDataRecordWithTags [Окно рецептов Синхронизировать запись данных с тегом]
- RecipeViewDeleteDataRecord [Окно рецептов Удалить запись данных]
- RecipeViewNewDataRecord [Окно рецептов Создать запись данных]
- RecipeViewGetDataRecordFromPLC [Окно рецептов Считать запись данных из ПЛК]
- RecipeViewRenameDataRecord [Окно рецептов Переименовать запись данных]
- RecipeViewShowInfoText [Окно рецептов Отобразить текст подсказки]
- RecipeViewMenu [Меню окна рецептов] (только для простого окна рецептов)
- RecipeViewOpen [Открыть окно рецептов] (только для простого окна рецептов)
- RecipeViewBack [Предыдущее окно рецептов](только для простого окна рецептов)

Системные функции для загрузки, сохранения и передачи записей данных рецептов и самих рецептов размещаются в группе "Recipes [Рецепты]".



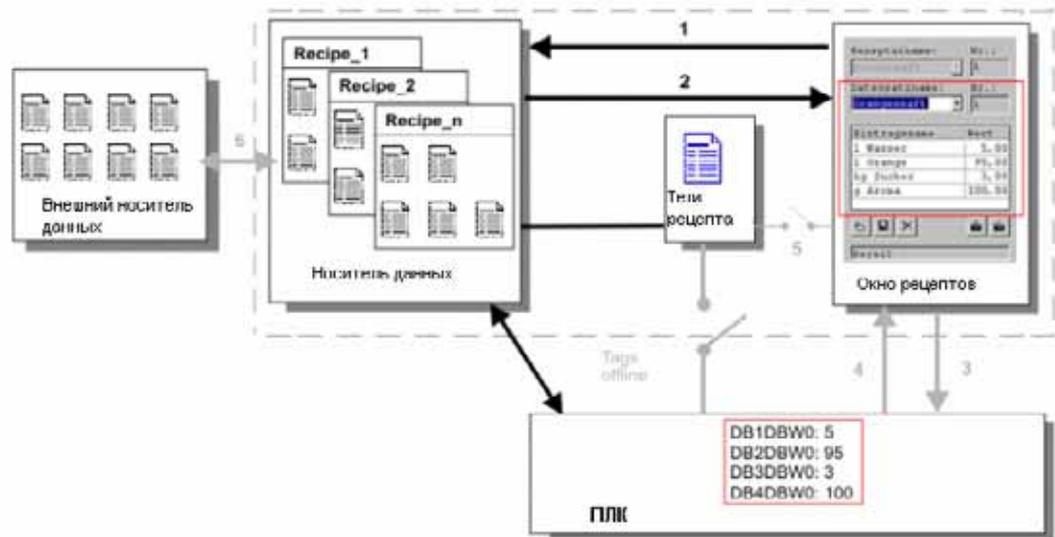
## 8.4 Сценарии управления производством с помощью рецептов

### 8.4.1 Сценарий: ввод производственных данных в режиме исполнения

#### Задача

Вы хотите ввести производственные данные на устройстве HMI, не вмешиваясь в исполняющийся технологический процесс. То есть, производственные данные не должны передаваться в ПЛК

## Решение задачи

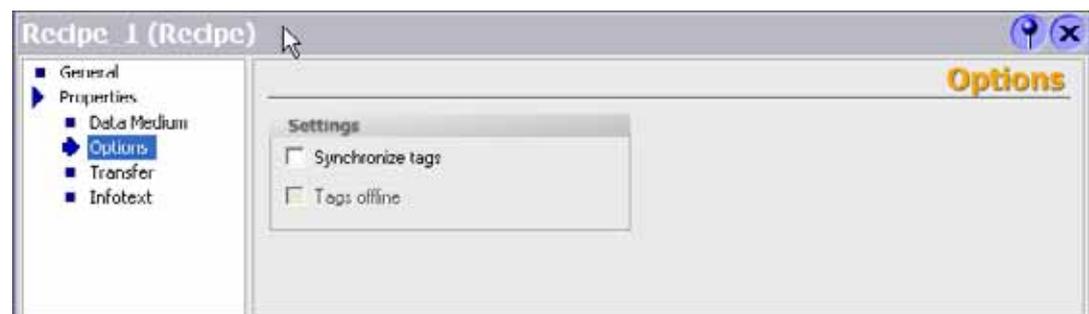


Вы вводите производственные данные в окне рецептов или на экране рецепта, определяете имя записи данных и сохраняете новую запись данных на носителе устройства HMI.

## Настройка в WinCC flexible

При создании рецепта указываются соответствующие теги.

Синхронизация с тегами рецепта не является необходимой, поскольку не предполагается передача данных производства (тегов) в ПЛК. Установите или сбросьте следующие флажки опций в окне свойств:



В зависимости от размера рецепта создайте окно рецептов или экран рецепта.

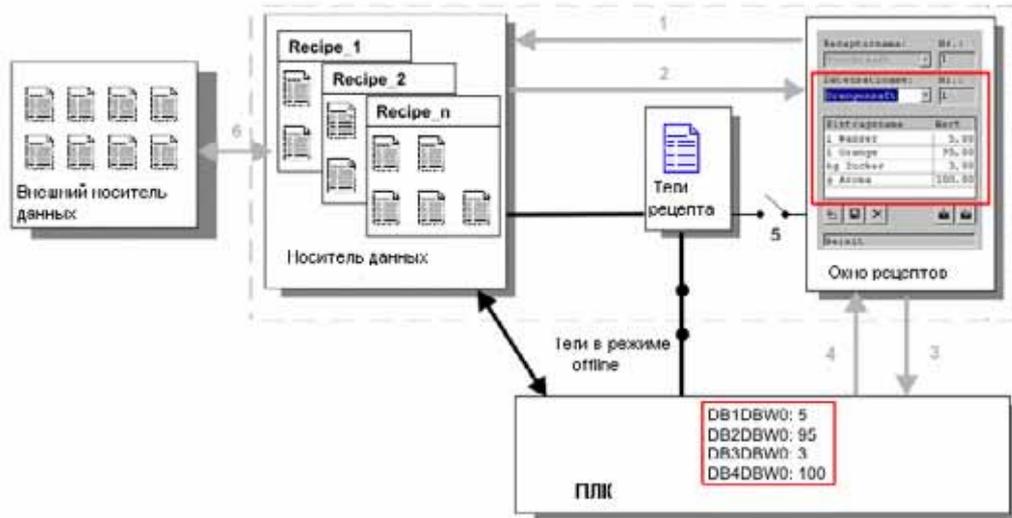
### 8.4.2 Сценарий: управление технологическим процессом путем корректировки производственных данных вручную

## Задача

Производственные данные, требуемые для выполнения предстоящих операций технологического процесса, будут запрашиваться с ПЛК и отображаться на устройстве

HMI с целью их проверки оператором. Вы хотите иметь возможность при необходимости вносить изменения в производственные данные в режиме online (обмена данными с контроллером).

### Решение задачи управления технологическим процессом

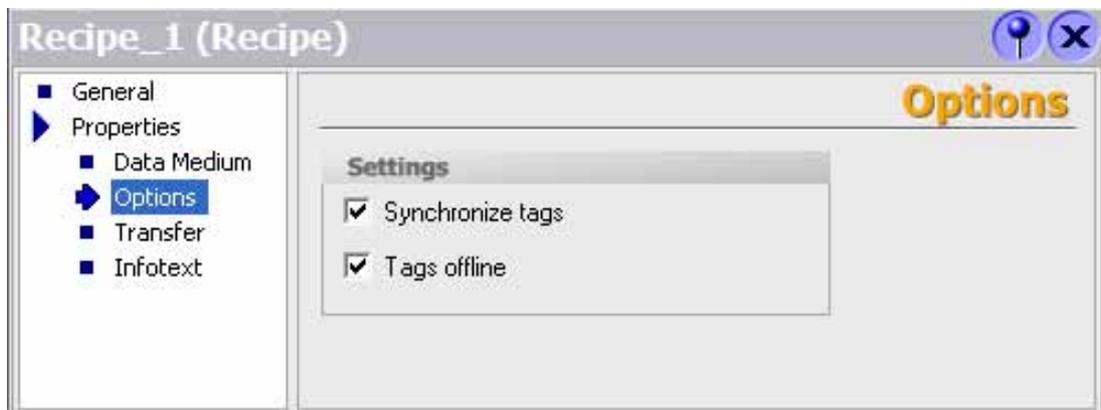


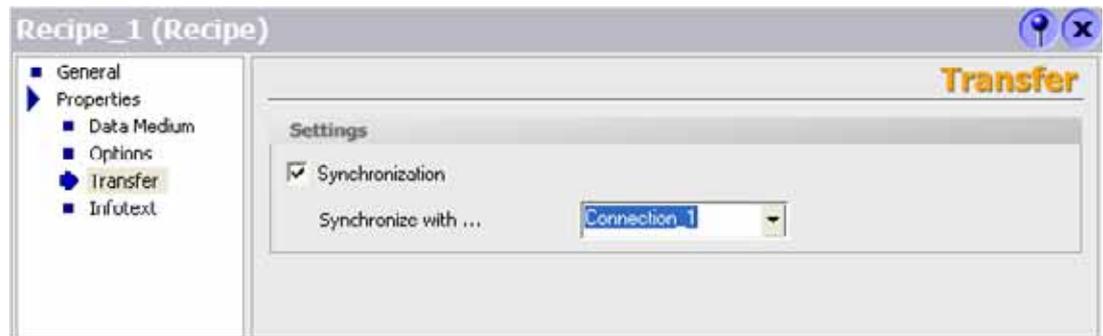
Устройство считывания, подключенное к ПЛК, считывает штрих-код операции технологического процесса, которая должна выполняться. Имена записей рецепта соответствуют штрих кодам операций. Это позволит ПЛК загрузить требуемые записи данных рецепта с носителя данных устройства HMI. Запись данных рецепта отображается на устройстве с целью ее проверки. Внесенные изменения немедленно передаются в ПЛК.

### Настройка в WinCC flexible

При создании рецепта указываются соответствующие теги.

Производственные данные будут передаваться в ПЛК, поэтому необходима их синхронизация с ПЛК с целью предотвращения случайной перезаписи этих данных. Теги должны передаваться в ПЛК. Выполните следующую настройку в окне свойств:





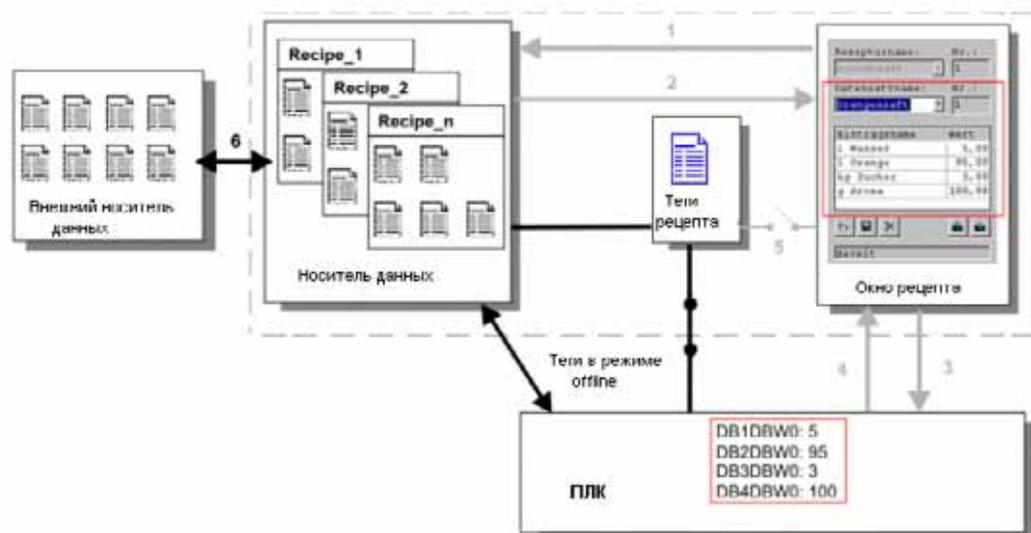
В зависимости от размера рецепта создайте окно рецептов или экран рецепта.

### 8.4.3 Сценарий: автоматическое управление технологическим процессом

#### Задача

Вы хотите, чтобы технологический процесс выполнялся автоматически без вмешательства оператора с целью корректировки производственных данных. Производственные данные должны передаваться непосредственно в ПЛК либо с носителя данных устройства HMI, либо с внешнего носителя данных. Производственные данные не должны отображаться на устройстве HMI.

#### Решение задачи управления технологическим процессом



Управление производством осуществляется путем выполнения одного или более скриптов ("Scripts"), которые автоматически передают производственные данные в ПЛК. Правильность выполнения может быть проконтролирована с помощью значений, возвращаемых используемыми функциями.

### **Настройка в WinCC flexible**

Процедура автоматического управления производством может быть реализована с помощью системных функций. Системная функция "ImportDataRecords" загружает записи данных рецепта из файла формата CSV на носитель данных. Системная функция "SetDataRecordTagsToPLC" передает записи данных рецепта с носителя данных в ПЛК.

## Архивирование и отображение тегов

### 9.1 Основные сведения об архивировании данных в WinCC flexible

#### 9.1.1 Основные сведения об архивировании данных технологического процесса

##### Введение

Операция архивирования используется для сбора, обработки и регистрации данных технологического процесса.

Эти данные впоследствии можно анализировать с целью получения важной технической и бизнес информации, касающейся рабочего состояния технологического оборудования.

##### Применение журналов данных

Процедура архивирования используется для документирования исполняющегося процесса и для получения журналов данных с целью последующего анализа ошибок и неисправностей. Путем анализа журналов данных вы можете получить информацию, которая позволит вам оптимизировать циклы профилактических осмотров и обслуживания, поднять качество производимой продукции и обеспечить ее соответствие установленным стандартам качества.

#### 9.1.2 Тренды

##### Введение

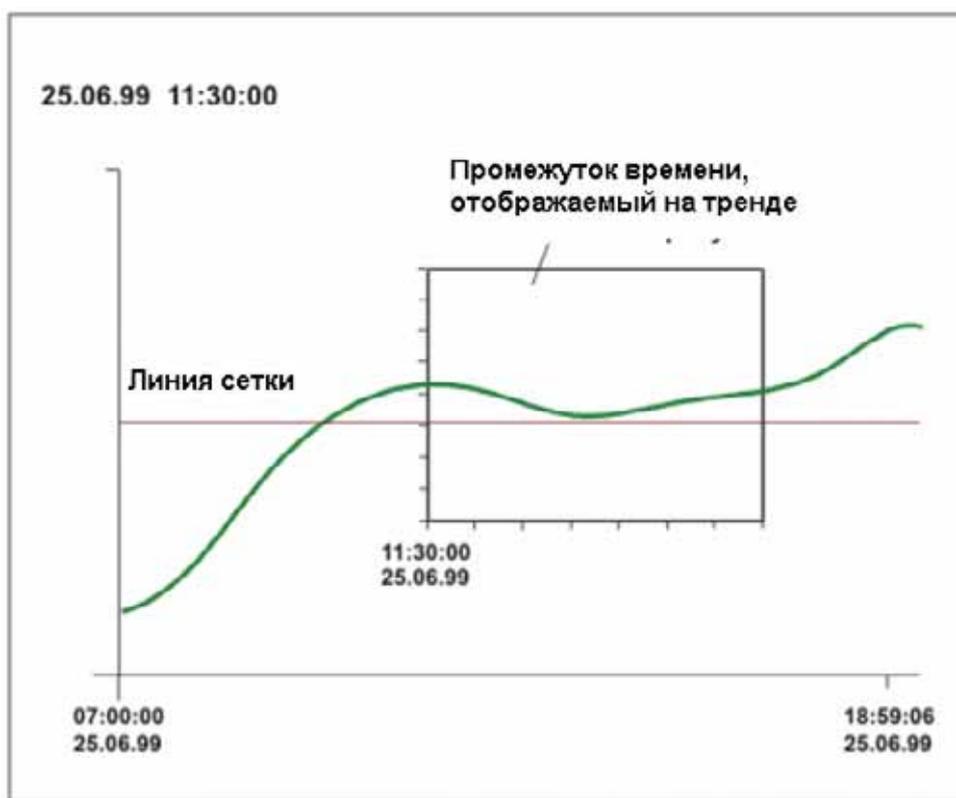
Тренд представляет собой графическое представление значений, которые принимает тег в режиме исполнения процесса. Для отображения трендов создайте на экране в вашем проекте окно трендов.

Для создания окна трендов укажите тип тренда отображаемых значений. Возможны следующие варианты:

- Журнальный тренд (англ. Log) - предназначен для отображения архивных или журнальных значений тега.
- Тренд значений, считываемых в определенные моменты реального времени, (англ. Realtime pulse triggered) - предназначен для отображения значений, считанных по времени.
- Тренд реального времени, считанный при установке бита, (англ. Realtime bit triggered) -предназначен для иницируемого событием отображения значений.
- Исторический тренд, считанный при установке бита, (англ. History bit-triggered) - предназначен для иницируемого событием отображения значений при использовании двух параллельных буферов

### Отображение журнальных значений

В окне тренда показаны журнальные значения в указанном промежутке времени. В режиме исполнения оператор может сдвигать временной период для того, чтобы посмотреть требуемую информацию (журнальные данные).



### Тренды значений, считываемых в определенные моменты реального времени

Отображаются значения, измеренные в определяемые соответствующим образом моменты времени. Тренды значений, считываемых по времени, используются для представления непрерывно протекающего процесса изменения некоторого параметра, например, изменения рабочей температуры двигателя.

## Тренды, считанные при установке бита

Отображение значений инициируется событием – установкой определенного бита в тегах "Trend transfer [инициализации передачи тренда]". По завершении операции чтения бит сбрасывается. Тренды, считывание значений которых из буфера тренда инициируется установкой бита, обычно используются для отображения быстро изменяющихся значений. Одним из примеров может быть давление впрыска при производстве пластиковых деталей.

## Тренды, считанные при установке бита, с использованием параллельного буфера для сбора данных

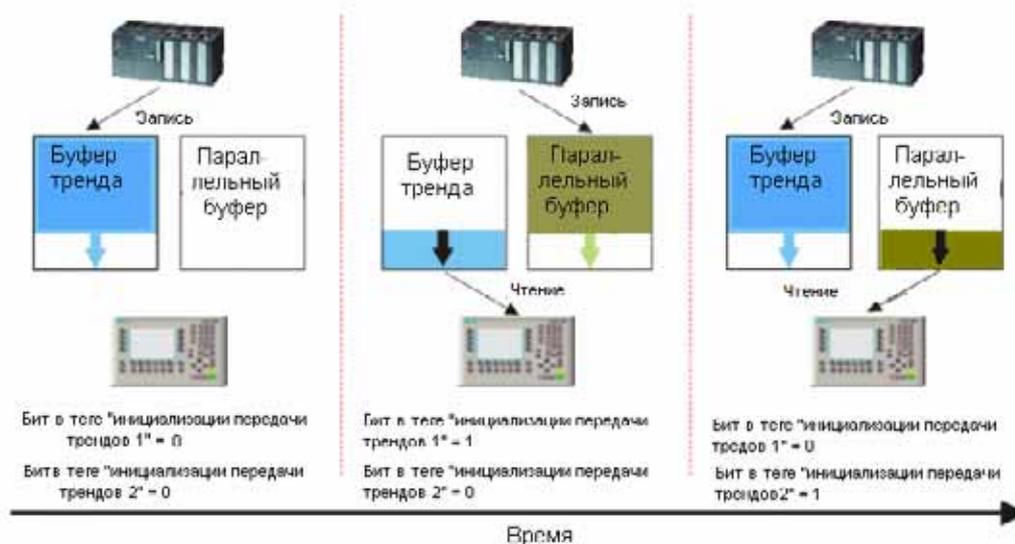
Если выбрана опция с использованием буферизации сбора данных, значения, которые должны будут отображаться на тренде, помещаются в буфер в ПЛК и считываются в виде блока данных при установке определенного бита. Такие тренды используются для отображения быстро изменяющихся значений, когда интерес представляет общий характер изменения значений тренда, а не отдельные его значения.

В ПЛК создается параллельный буфер или буфер переключения (англ. switch buffer). Пока идет выполнение операции чтения буфера тренда, новые значения процесса записываются в этот параллельный буфер. Параллельный буфер гарантирует, что новые измеренные значения не будут записаны в ПЛК на место старых, пока операторское устройство считывает значения, которые будут отображаться в виде тренда.

Переключение между буфером тренда и параллельным буфером происходит следующим образом:

Всякий раз, когда в теге "Trend transfer 1 [инициализации передачи трендов 1]" установлен бит, соответствующий тренду, все значения одновременно считываются из буфера тренда и отображаются на операторском устройстве в виде тренда. По завершении процедуры считывания бит в области "Trend Transfer 1 [Инициализации передачи трендов 1]" сбрасывается.

Пока операторское устройство считывает значения тега из буфера тренда, ПЛК записывает новые значения в параллельный буфер. Когда в теге "Trend transfer 2 [инициализации передачи трендов 2]" устанавливается бит, соответствующий тренду, все значения считываются из параллельного буфера и отображаются на операторском устройстве в виде тренда. Пока операторское устройство читает параллельный буфер, новые значения записываются в ПЛК опять в буфер тренда.



### 9.1.3 Архивирование данных в WinCC flexible

#### Введение

Данными называют информацию, собираемую в ходе технологического процесса и сохраняемую в памяти одного из подключенных к системе контроллеров. Эти данные отражают состояние технологического оборудования, например, значения температуры, уровни наполнения или другие состояния (например, двигатель выключен). Для работы с переменными процесса необходимо определить теги в WinCC flexible.

Внешние теги используются в WinCC flexible для сбора значений процесса и для доступа к областям памяти подключенного контроллера (системы автоматизации). Внутренние теги не имеют связи с процессом и доступны только на соответствующем устройстве HMI.

#### Основные сведения

Значения внутренних и внешних тегов могут быть сохранены в журналах данных. Вы можете отдельно определить журнал, в котором будет сохранен каждый тег.

Процессом архивирования данных в журналах можно управлять через циклы и события. Циклы архивирования используются для непрерывного сбора и сохранения значений тегов. Кроме того, процедура архивирования может быть инициирована событием, например, изменением значения. Такая настройка может быть сделана отдельно для каждого тега.

В режиме исполнения архивируемые значения тегов измеряются, обрабатываются и записываются в базу данных ODBC или файл.

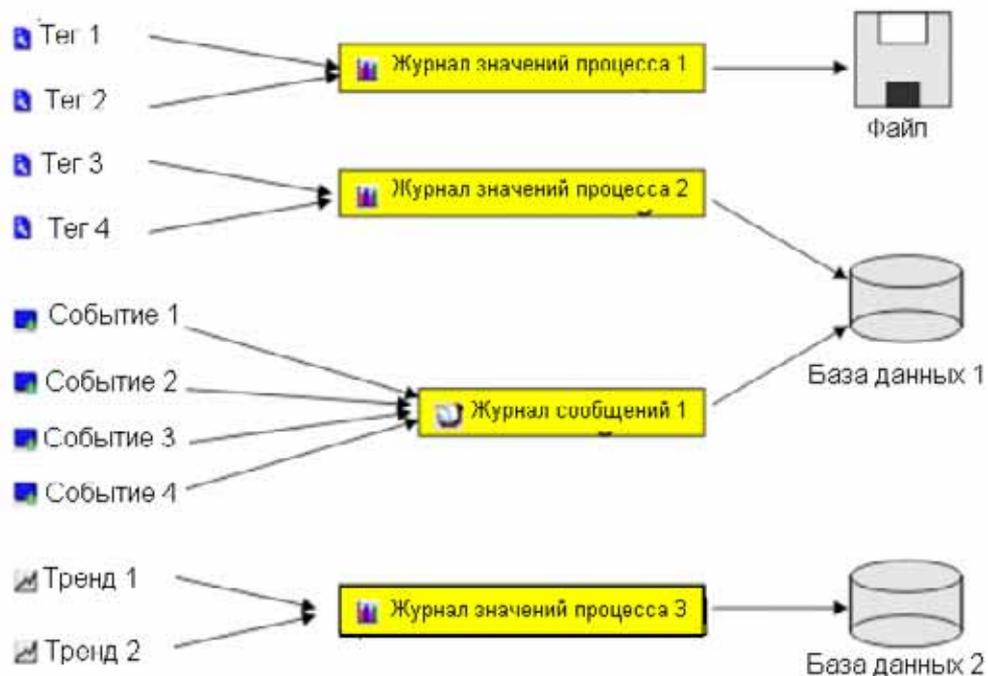
#### Типы журналов

WinCC flexible позволяет выбрать один из следующих типов журналов:

- Циклический журнал
- Сегментированный циклический журнал
- Циклический журнал с отправкой сообщения при его заполнении
- Циклический журнал с вызовом функции при его заполнении.

### Носитель данных и память для хранения журналов

Архивируемые данные будут сохранены либо в базе данных ODBC (только на ПК) либо в файле.



В зависимости от конфигурации аппаратного обеспечения и параметров настройки устройства HMI device, данные могут архивироваться локально (на жестком диске ПК или на карте памяти панели) или, если таковой имеется, на сетевом дисковом накопителе.

Сохраненные данные могут быть обработаны другими программами, например, с целью их анализа.

### Вывод содержимого журнала

В режиме исполнения журнальные значения тегов могут быть выведены на экранах процессов в виде трендов.

## 9.2 Элементы пользовательского интерфейса и основная настройка

### 9.2.1 Редактор журналов данных "Data Logs"

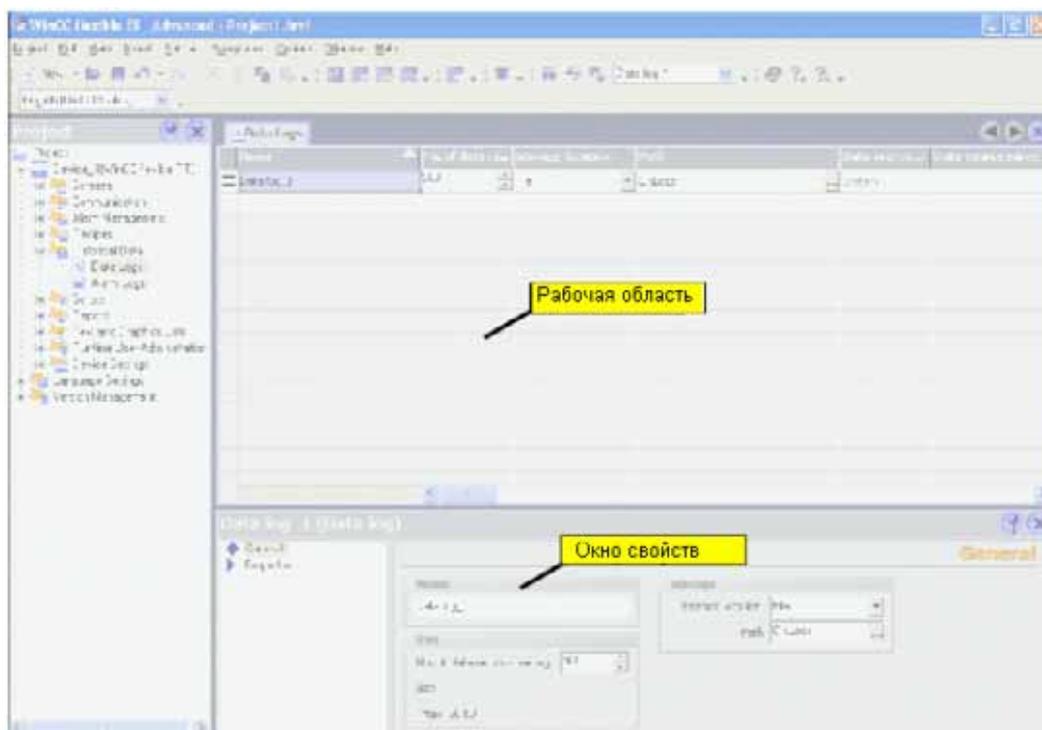
#### Введение

Для архивирования значений процесса необходимо определить журнал. Создание журнала и определение его свойств выполняется в редакторе журналов данных "Data Log".

#### Открытие редактора

Для открытия редактора журналов щелкните дважды на элементе " Data Log [Журнал данных]" в группе "Logs [Журналы]".

#### Пользовательский интерфейс редактора



#### Рабочая область

Все журналы данных отображаются в таблице в рабочей области. Свойства журналов данных можно редактировать непосредственно в ячейках таблицы. Таблицу можно отсортировать по записям столбца, щелкнув мышью на заголовке этого столбца.

Столбцы таблицы можно скрыть и, наоборот, отобразить. Для этого активизируйте или деактивируйте соответствующие пункты во всплывающем меню заголовка столбца таблицы.

## Окно свойств

Здесь задаются параметры журналов данных. В окне свойств отображается та же информация и параметры, что и в таблице рабочей области.

## 9.2.2 Определение основных свойств журналов данных

### Введение

Свойства журналов данных могут быть определены в редакторе журналов данных или в окне свойств журналов.

### Структура окна свойств

Слева в окне свойств отображается иерархическая структура, в которой можно выбирать различные категории свойств. Поля для определения значений свойств выбранной категории показаны в правой части окна свойств.

В окне свойств можно указать следующие свойства журналов данных:

### Общие свойства

- Name [Имя]  
Журнал данных может иметь любое имя, однако имя журнала должно содержать, по крайней мере, одну букву или цифру.
- Область памяти для хранения журнала  
Журнал данных храниться в базе данных ODBC (только на ПК) или в отдельном "\*.csv"-файле. Выберите в качестве места хранения "File [Файл]" или "Database [База данных]" соответственно.  
Для определения пути к месту хранения ("path") вы можете, в зависимости от устройства HMI выбрать жесткий диск ПК или карту памяти панели, или, если возможно, сетевой дисковый накопитель.  
Если в качестве места хранения выбрана база данных ODBC, вы можете использовать имя, предлагаемое системой (системное имя источника данных) или ввести свое имя (пользовательское имя источника данных).
- Size [Размер]  
Размер журнала вычисляется следующим образом:  
количество элементов \* длину каждого значения тега, которое будет занесено в журнал.  
Минимальный и максимальный размер журнала с выбранным в данный момент количеством записей данных показан под полем ввода "Number of data records"

[Количество записей данных]" в окне свойств. Максимальный размер ограничен объемом доступной памяти на устройстве НМІ.

### Настройка функционирования журнала

- Запуск процедуры архивирования

С помощью соответствующих параметров можно настроить процедуру архивирования таким образом, что архивирование данных начнется при запуске системы исполнения проекта. С этой целью установите флажок опции "Начать архивирование при запуске системы исполнения."

Вы можете также выполнить настройку начала записи данных в журнал при запуске режима исполнения другими способами. Установите флажок опции "Reset log [Очистить журнал]", если вы хотите заменить данные, записанные в журнал ранее, новыми данными или опции "Append data to existing log [Добавить данные в существующий журнал]", если вы хотите добавить новые данные к данным, уже записанным в журнал.

---

#### Замечание

Для очистки журнала в режиме исполнения могут быть использованы системные функции.

---

- Метод архивирования

Здесь определяется, что происходит с журналом, когда он полон. Вы можете выбрать один из следующих вариантов:

- Циклический журнал: При заполнении журнала новые данные записываются на место самых старых записей.
- Сегментированный циклический журнал: Создается несколько журналов одного и того же размера. Журналы заполняются данными. Когда полностью заполнены все журналы, новые данные заносятся в журнал, который был заполнен первым.
- Вывод системного сообщения: при достижении определенного уровня заполнения появляется системное сообщение.
- Генерирование события: Как только журнал заполнен, генерируется событие Overflow [Переполнение].

- "Comment [Комментарий]"

Комментарий содержит описание журнала.

### События

Здесь определяется список функций, которые будут выполняться при возникновении события переполнения, генерируемом заполнением журнала.

## 9.3 Архивирование значений тегов

### Введение

При исполнении проекта значения тегов могут записываться в журналы с целью их анализа на более позднем этапе. Для архивирования тега необходимо определить журнал, в который будут заноситься значения тега, частоту выполнения этих операций и режим архивирования, определяющий, будут ли заноситься в журнал только значения тега из указанного диапазона, или все значения.

---

#### Замечание

Основной целью архивирования является запись в журнал значений внешних тегов. Однако можно архивировать и значения внутренних тегов.

---

### Основные сведения

Для настройки процедуры архивирования данных необходимо выполнение ряда действий:

- Создание и определение параметров (свойств) журналов данных  
При создании журнала данных необходимо определить следующее:
  - общие параметры настройки, например, имя, размер, область памяти для хранения,
  - функционирование при запуске среды исполнения,
  - функционирование при заполнении журнала
- Настройка процедуры архивирования тегов  
Журнал можно определить для каждого тега. В такой журнал в режиме исполнения записываются значения тега и другая информация, например, время регистрации значения.  
Кроме того, вы можете определить, в какие моменты и как часто значения тега будут заноситься в журнал. При определении этих параметров возможны следующие варианты:
  - "On request [По запросу]":  
Значения тега заносятся в журнал при вызове системной функции "LogTag".
  - "On change [При изменении значения]"  
Значение тега заносится в журнал, как только на операторском устройстве обнаружено изменение значения тега.
  - Cyclic continuous [Постоянное циклическое]  
Значения тега заносятся в журнал через регулярные интервалы времени. В дополнение к стандартным циклам, существующим в WinCC flexible, вы можете определить свои циклы, значения времени которых основаны на стандартных.  
Кроме того, можно архивировать только те значения, которые находятся в пределах допустимого диапазона. Таким образом, вы можете записывать значения тега в различные журналы с целью последующего анализа.

## 9.4 Вывод журнальных данных

- Последующая обработка значений тегов, занесенных в журналы  
Анализ значений тегов, занесенных в журналы, можно выполнить непосредственно в проекте WinCC flexible, например, в окне трендов или с помощью другого приложения, например, Excel.

## 9.4 Вывод журнальных данных

### 9.4.1 Вывод значений тега на экранах

#### Введение

В режиме исполнения значения тега могут быть выведены на экране операторского устройства в виде тренда. Данные процесса могут быть получены из ПЛК или загружены из базы данных журнала.

#### Отображаемые значения

Для отображения значений тега на операторском устройстве необходимо на экране процесса сконфигурировать окно трендов. При настройке окна трендов определяется, какие значения будут отображаться на экране операторского устройства:

- Текущие значения из ПЛК  
На тренде отображаются либо отдельные значения из ПЛК (отображение в реальном времени), либо все значения, сохраненные в буферной области памяти (буфере) между двумя операциями чтения буфера в ПЛК (отображение значений за интервал времени).  
Операция чтения буфера может происходить в моменты времени, определяемые установкой бита в области инициализации передачи тренда, либо циклически.
- Журнальные значения тегов  
В режиме исполнения значения тегов из соответствующих журналов данных отображаются в окне трендов. На тренде показаны журнальные значения в указанном временном интервале. В режиме исполнения оператор может сдвигать временной интервал для того, чтобы посмотреть требуемую информацию за другой период времени (журнальные данные).

### 9.4.2 Структура \*.csv файла с тегами

#### Введение

В файле формата \*.csv (Comma separated value [значения, разделяемые запятыми]), столбцы таблицы (имя и значение записи) отделяются точкой с запятой. Каждая строка таблицы заканчивается возвратом каретки.

**Пример файла формата \*.csv**

В следующем примере показан файл со значениями тега:

```
"VarName";"TimeString";"VarValue";"Validity";"Time_ms"
"Var_107";"01.04.98 11:02:52";66,00;1;35886460322,81
"Var_107";"01.04.98 11:02:55 AM";60.00;1;35886460358.73
"Var_107";"01.04.98 11:02:57 AM";59.00;1;35886460381.22
```

**Структура файла журнала в формате \*.csv**

Следующие значения вводятся в отдельные столбцы журнального файла WinCC flexible:

Параметр	Описание
VarName	Имя тега WinCC flexible
Time string	Временная метка в виде STRING [СТРОКИ], например, дата в читаемом формате
VarValue	Значение тега
Validity	Истинность: 1 = значение истинно 0 = произошла ошибка (например прервано соединение с процессом)
Time_ms	Временная метка в виде десятичного значения (преобразование данных см ниже). Необходима только для отображения значений тега на тренде.

**Преобразование десятичного значения временной метки**

Если значение необходимо преобразовать с помощью другой программы, выполните следующие действия:

1. Разделите значение Time\_ms на 1,000,000.

Пример: 36343476928:1 000 000 = 36343,476928

2. Целая часть числа (36344) будет являться датой, отсчитанной от 31.12.1899.

Пример: 36343 соответствует 02.07.1999

Теперь вы можете преобразовать значение временной метки в дни в программе Excel путем назначения ячейкам, содержащим временную метку, соответствующего формата из группы "Date [Дата]".

Результат: 37986 соответствует 31.12.2003

3. Значение после запятой (0,476928) показывает время:

- Путем умножения значения (0,476928) на 24 получаются часы (11,446272).
- Путем умножения остатка (0,446272) на 60 получаются минуты (26,77632).
- Путем умножения остатка (0,77632) на 60 получаются секунды (46,5792).

Сумма 11:26:46.579

Такое преобразование может быть выполнено, например, в Microsoft Excel

### 9.4.3 Непосредственное обращение к журнальной базе данных ODBC

#### Введение

Журнал может быть организован в виде базы данных или файла.

Обращение к базе данных происходит по ее имени Data source name (DSN) Выберите базу данных, которую вы хотели бы использовать в WinCC flexible с помощью меню Пуск, выбрав команды Settings > Control panel > ODBC data sources [Настройка > Панель управления > Источник данных ODBC].

Для сохранения данных журнала определите "Data source name [Имя источника данных]" (DSN) вместо имени директории при разработке проекта. С помощью DSN вы обращаетесь к базе данных и к месту ее хранения.

#### Применение

Для дополнительной обработки и анализа данных журнала можно использовать весь спектр возможностей работы с базами данных.

#### Основные сведения

Вы создаете источник данных, который связан с базой данных на том же компьютере, на котором содержится программное обеспечение среды исполнения. Имя созданного источника данных DSN указывается при создании журнала в WinCC flexible.

С помощью интерфейса ODBC вы можете непосредственно обращаться к этой базе данных из других программ, таких как MS Access или MS SQL server.

Используя системную функцию "StartProgram [Запустить программу]" вы можете настроить вызов программы с устройства HMI (например, для приложения MS Access). При этом последовательность вызова программ системы исполнения не нарушается.





## Работа с системой отчетов

### 10.1 Основные сведения о системе отчетов

#### Введение

В WinCC flexible для документирования данных процесса и данных завершенных производственных циклов используются отчеты. Отчеты сообщений и данных рецептов могут использоваться для создания отчетов рабочих смен, вывода данных о партиях продуктов или для того, чтобы документировать данные производственного процесса для приемочной комиссии.

#### Обзор

Файлы отчетов можно редактировать в графическом редакторе. Редактор позволяет конфигурировать структуру отчета и определять данные, которые будут выводиться в отчете. Для вывода данных в файле отчета вы можете добавлять различные объекты. В качестве таких объектов могут использоваться объекты окна инструментов. При этом некоторые из них используются либо с ограничениями, либо совсем недоступны. Это зависит от используемого устройства HMI. Недоступные объекты отображаются серыми и не могут быть выбраны оператором.

Для вывода различных типов данных вы можете создать разные файлы отчетов. Кроме того, вы можете настроить запуск процедуры вывода отчета отдельно для каждого файла отчета. Можно, например, выбрать вариант запуска процедуры вывода в определенный момент времени или через указанный интервалы времени, либо указать события, инициирующие вывод отчета.

Модульная структура этих функций позволяет конфигурировать отчеты в соответствии с конкретными требованиями пользователя.

#### Примеры применения

Можно создать отчет, генерируемый в конце смены, который будет содержать данные о произведенной партии продукта и ошибках, которые относятся к уже завершенному производству.

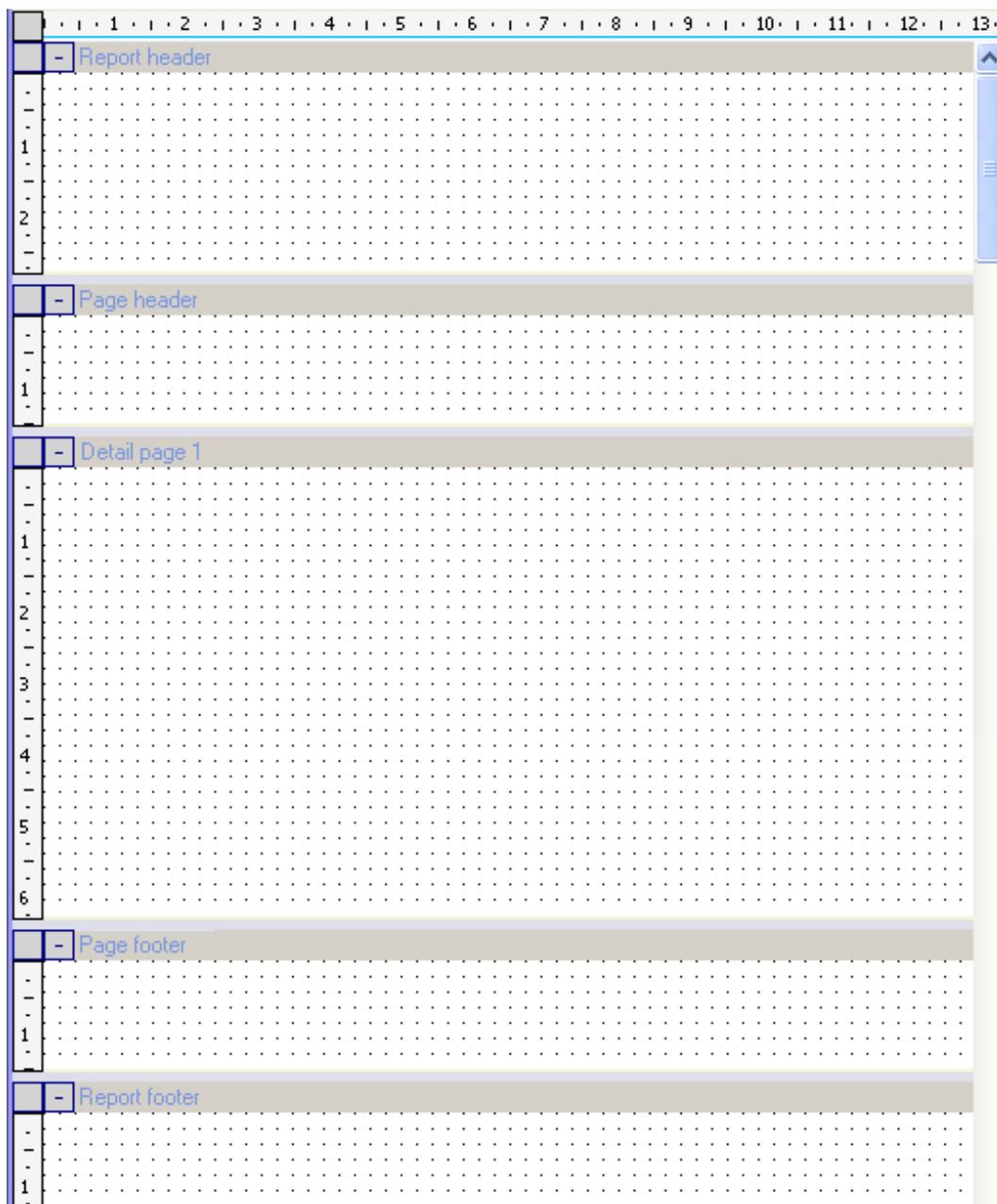
Можно создать отчет, в котором будут выводиться производственные данные серийного производства.

Еще одним примером отчета может быть отчет, в котором выводятся сообщения определенного класса или типа.

## 10.2 Структура отчета

### Структура отчета

Все отчеты в WinCC flexible имеют одинаковую структуру. Они делятся на различные секции, как показано на рисунке, приведенном ниже.



Разные секции отчета используются для вывода различных данных и могут содержать общие объекты и специфические для конкретного отчета объекты.

- **Заголовок отчета**  
Заголовок отчета является титульным листом отчета. Заголовок отчета служит для вывода названия проекта и общей информации о проекте. Заголовок отчета выводится без заголовка страницы и без нижнего колонтитула. Заголовок отчета выводится только один раз в начале отчета.
- **Заключение отчета**  
Заключение отчета является последней страницей отчета. Заключение отчета служит для вывода краткой итоговой информации отчета или другой информации, необходимой в конце отчета. Заключение отчета выводится без заголовка страницы и без нижнего колонтитула. Заключение отчета выводится только один раз в конце отчета.
- **Заголовок страницы (верхний колонтитул)**  
Заголовок страницы выводится на каждой странице отчета. Заголовок страницы служит для вывода даты, времени, названия или другой информации общего характера.
- **Нижний колонтитул страницы**  
Нижний колонтитул страницы выводится на каждой странице отчета. Нижний колонтитул служит для вывода номера страницы, общего количества страниц отчета или другой информации общего характера.
- **Страница данных отчета (англ. detail page)**  
Данные режима исполнения выводятся в области "Detail page [Страница данных отчета]". Объекты для вывода данных режима исполнения помещаются в области "Detail page [Страница данных отчета]". При выводе данных в зависимости от их количества автоматически добавляются разрывы страниц. Для того чтобы оптически разделить различные объекты вывода вы можете вставить в отчет несколько страниц.

Процедура создания отчета описана в главе "Создание отчета".

## 10.3 Элементы пользовательского интерфейса и основная настройка

### 10.3.1 Редактор отчетов

#### Введение

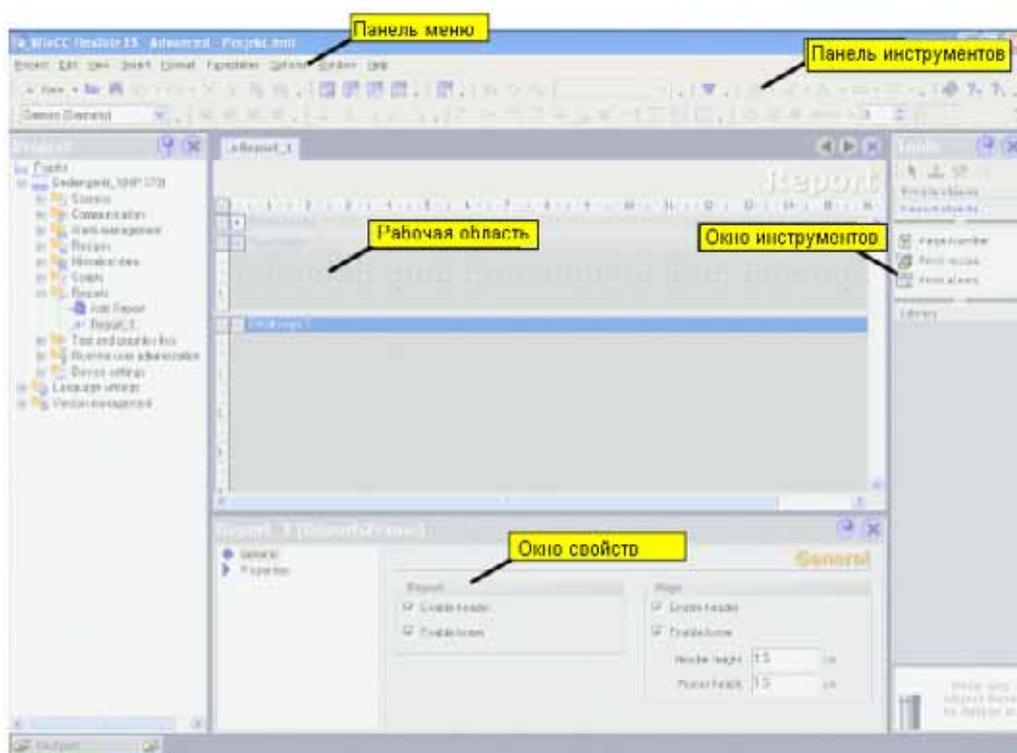
Создание и редактирование отчетов выполняется в редакторе отчетов.

### Открытие редактора

Выберите элемент "Report [Отчет]" в окне проекта и откройте контекстное меню. Выберите во всплывающем меню пункт "New report [Новый отчет]". Новый отчет будет создан и открыт в рабочей области.

Для того чтобы открыть существующий отчет дважды щелкните на требуемом отчете в окне объектов. Будет открыт выбранный отчет.

### Пользовательский интерфейс редактора



### Панель меню

Панель меню содержит все команды, необходимые для работы WinCC flexible. Клавиши быстрого вызова команд показаны рядом с командами меню.

### Панели инструментов

Панели инструментов содержат наиболее часто используемые кнопки.

С помощью команды меню "View > Toolbars [Вид > Панели инструментов]" вы можете отобразить или скрыть имеющиеся панели инструментов. Кнопка панели инструментов используется для отображения или скрытия отдельных кнопок этой панели.

## Рабочая область

В рабочей области выполняется конфигурирование отчетов.

## Окно инструментов

Окно инструментов предоставляет доступ к объектам, требуемым для конфигурирования отчета. Объекты вставляются в отчет с помощью функции перетаскивания.

## Окно свойств

При выборе некоторого объекта в окне свойств можно редактировать свойства этого объекта.

Если не выделен ни один объект, в окне свойств можно редактировать свойства активной области отчета.

### 10.3.2 Использование окна инструментов

#### Введение

Окно инструментов содержит набор объектов, объединенных в группы "Simple objects [Простые объекты]" и "Report objects [Объекты отчета]", которые вы можете вставлять в ваши отчеты.

Если в рабочей области отображается отчет, в окне инструментов содержатся только те объекты, которые могут быть использованы в отчете. Некоторые из объектов окна инструментов используются либо с ограниченными функциональными возможностями, либо совсем недоступны. Это зависит от используемого устройства HMI. Недоступные объекты отображаются серыми и не могут быть выбраны оператором.

#### Изменение свойств, определенных по умолчанию

Для различных объектов окна инструментов определены свойства по умолчанию. Объект, вставляемый из окна инструментов в отчет обладает этими предварительно определенными свойствами.

Вы можете изменить эти свойства в соответствии с требованиями вашего проекта. При изменении свойств объекта определенного типа, свойства объектов, уже вставленных в отчет, сохраняются. Поэтому, следует изменять свойства перед тем, как объект вставлен в отчет.

Свойства объекта, заданные по умолчанию, привязаны к имени, под которым вы зарегистрированы в системе.

Для изменения свойств объекта, заданных по умолчанию, следует открыть всплывающее меню объекта в окне инструментов. Выберите команду "Edit default properties [Редактировать свойства по умолчанию]" Отображается диалоговое окно свойств. Измените свойства объекта в соответствии с требованиями вашего проекта.

## Отображение окна инструментов

С помощью команды меню "View > Toolbox view [Вид > Окно инструментов]" вы можете отобразить или скрыть окно инструментов.

## 10.4 Работа с отчетами

### 10.4.1 Создание отчета

#### Введение

При создании отчета определяется содержимое следующих секций отчета:

- заголовка отчета,
- заголовка страницы (верхнего колонтитула),
- страницы отчета,
- нижнего колонтитула страницы,
- нижнего колонтитула отчета.

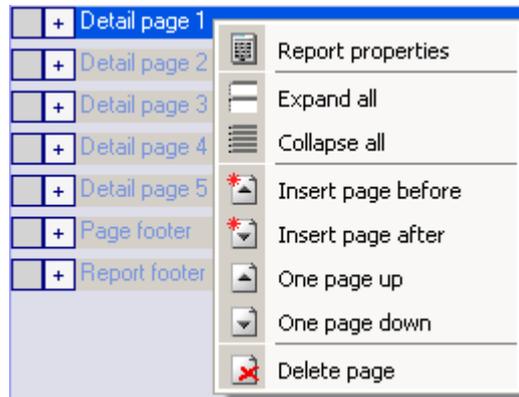
#### Обзор процедуры создания отчета

Для создания отчета и определения данных для вывода в отчете могут использоваться объекты окна инструментов. Некоторые объекты имеют ограниченный набор функций по сравнению с их функциями при использовании этих же объектов для создания экранов. Например, поле ввода/вывода может служить только полем вывода.

При создании отчета в редакторе отчетов, он отображается по принципу wysiwyg – то, что видишь на экране, то и получишь при печати. Динамические объекты для вывода данных, например, "Print alarm [Печать сообщения]" и "Print recipe [Печать рецепта]" являются исключениями. Определяемый при проектировании вертикальный размер этих объектов не связан с форматом вывода, поскольку размер динамических объектов зависит от объема выводимых данных. Разрывы страниц вставляются на страницах отчета последовательно, в зависимости от объема данных. На каждую страницу отчета можно вставить только один из этих объектов. Объекты, помещаемые на той же стороне, ниже одного из этих объектов не выводятся. Объекты "Print alarm [Печать сообщения]" и "Print recipe [Печать рецепта]" вставляются автоматически, причем их горизонтальный размер определяется горизонтальным размером, указанным для отчета. Горизонтальный размер при выводе отчета определяется указанным горизонтальным размером отчета.

Новый отчет всегда содержит только одну страницу. Эта страница представляет собой страницу выводимых данных отчета. При необходимости в отчет можно вставить дополнительные страницы. С этой целью переместите курсор на строку заголовка уже имеющейся страницы и откройте с помощью правой кнопки мыши контекстное меню. Для вставки страницы до или после уже имеющейся страницы отчета необходимо использовать команды "Insert page before [Вставить страницу до]" и "Insert page after [вставить страницу после]". Страницам присваиваются последовательные номера. В отчете может быть не более 10 страниц. При создании более 10 страниц номера

лишних страниц проставляются заключенными в угловые скобки (например: Стр <11>). Лишние страницы при выводе не учитываются. Для удаления выделенной страницы используется команда всплывающего меню "Delete page [Удалить страницу]".



Последовательность страниц можно изменить. С этой целью переместите курсор на нужную страницу и откройте с помощью правой кнопки мыши контекстное меню. Выберите во всплывающем меню команду "One page up [На страницу вверх]" или "One page down [На страницу вниз]". Страница соответствующим образом переместится. Последовательная нумерация страниц сохранится. Если, например, страница 4 перемещена с помощью команды "One page up [На страницу вверх]", в результате страницы 3 и 4 поменяются местами.

Для того чтобы получить более наглядный обзор в рабочей области, можно закрыть отдельные секции отчета. С этой целью щелкните на соответствующем узле рядом с обозначением секции.



Кроме того, вы можете отобразить или скрыть все области одновременно. С этой целью переместите курсор на строку заголовка области отчета и откройте с помощью правой кнопки мыши контекстное меню. Выберите в меню команду "Display all [Отобразить все]" или "Hide all [Скрыть все]".



## 10.4.2 Редактирование свойств отчета

### Введение

При редактировании свойств отчета выбираются варианты вывода и формата. Существуют следующие группы свойств:

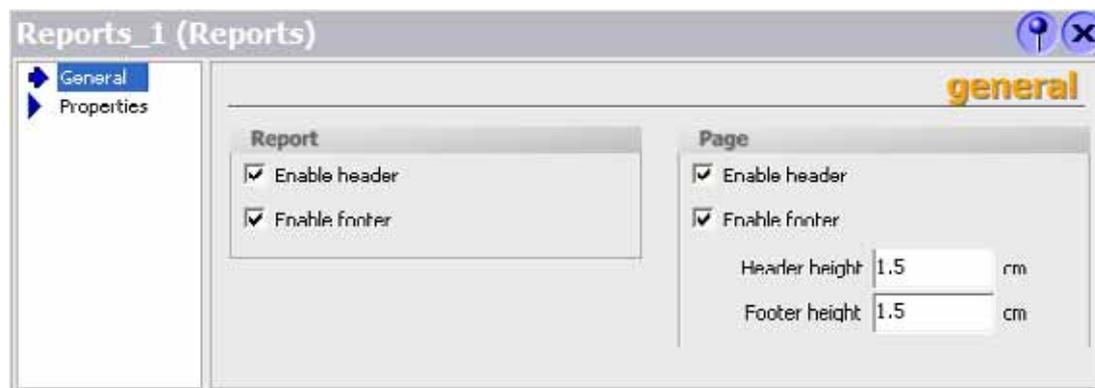
- Общие
- Свойства, касающиеся формата вывода

### Предварительные условия

- Отчет, свойства которого необходимо изменить, должен быть открыт.
- Должно быть открыто окно свойств.

### Редактирование свойств отчета

Для редактирования свойств отчета поместите курсор на строке заголовка соответствующей области отчета, например, на строке заголовка области "Page header [Заголовок страницы]". С помощью правой кнопки мыши откройте всплывающее меню и выберите команду "Document properties [Свойства документа]". Свойства отчета отображаются в окне свойств.



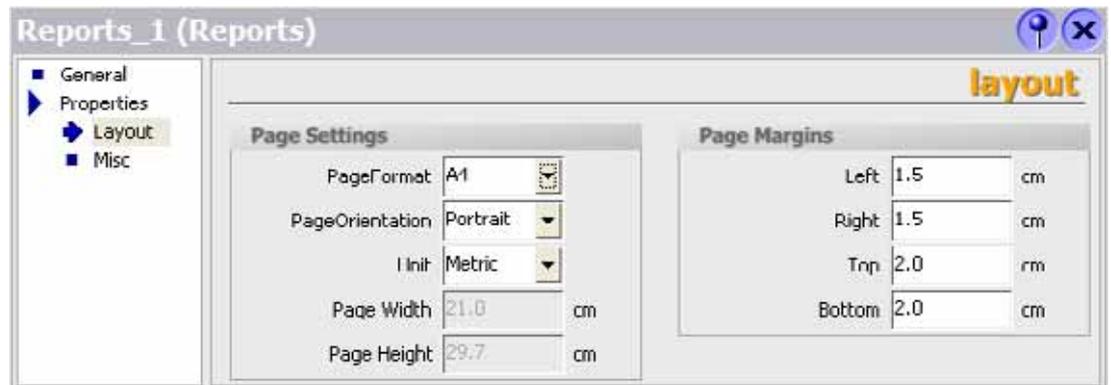
Выберите в окне свойств группу "General [Общие]".

В области "Report [Отчет]" установите или снимите флажок опции вывода заголовка отчета.

В области "Report [Отчет]" установите или снимите флажок опции вывода заголовка отчета.

Если вывод какой-либо области отчета отменен указанным способом, то в строке заголовка этой области будет выведен символ "(X)".

Выберите в окне свойств группу "Properties/Layout [Свойства/Макет]".



В поле "Page Format [Формат страницы]" выберите формат выводимой страницы.

В качестве альтернативы можно выбрать "User defined [Пользовательский формат]". При этом необходимо будет ввести значения, определяющие пользовательский формат вывода, в поля "Page Width [Горизонтальный размер]" и "Page Height [Вертикальный размер]".

В поле "Page Orientation [Ориентация страницы]" выберите книжную или альбомную ориентацию.

В поле "Unit [Единица измерения]" выберите единицу измерения для определения размера и полей страницы.

Укажите размер полей страницы в полях области "Page Margins [Поля страницы]". Поля, определяемые в этой области, не должны быть меньше полей, определенных для принтера.

### 10.4.3 Объекты для создания отчета

#### Введение

Объекты могут представлять собой либо графические элементы для создания структуры отчета, либо динамические элементы для вывода данных. Объекты имеют ограничения, которые зависят от используемого устройства НМІ. Обращайте внимание на информацию в описаниях объектов.

Объекты можно найти в группе объектов "Simple objects [Простые объекты]" в окне инструментов.

Для создания отчетов можно также использовать специальные объекты отчета. Эти объекты находятся в группе "Report objects [Объекты отчета]".

#### Простые объекты

Символ	Объект	Описание
	Линия	Линия является незамкнутым объектом. Длина линии и угол наклона определяются высотой и шириной прямоугольника, содержащего объект. Линия может заканчиваться стрелками или точками.

Символ	Объект	Описание
	Ломаная	Ломаная линия является незамкнутым объектом. Область, ограниченная линией не может быть залита (цветом или узором) даже в том случае, если начальная и конечная точки имеют одинаковые координаты. Ломаная линия может иметь любое количество вершин. Вершины нумеруются в порядке их создания и могут быть изменены или удалены пол-отдельности. Ломаная линия может заканчиваться стрелками или точками.
	Многоугольник	Многоугольник является замкнутым объектом и может быть залит цветом или узором. Многоугольник может иметь любое количество вершин. Вершины нумеруются в порядке их создания и могут быть изменены или удалены пол-отдельности.
	Эллипс	Эллипс является замкнутым объектом и может быть залит цветом или узором. Для выравнивания объекта по горизонтали или вертикали вы можете задавать ширину или высоту эллипса.
	Окружность	Окружность является замкнутым объектом и может быть залита цветом или узором. Диаметр окружности может быть задан произвольно.
	Прямоугольник	Прямоугольник является замкнутым объектом и может быть залит цветом или узором. Длина и ширина прямоугольника могут быть заданы произвольно в соответствии с требованиями пользователя. При необходимости вершины прямоугольника могут быть скруглены.
	"Текстовое поле	Поле статического текста является замкнутым объектом и может быть залито цветом или узором. Статический текст может быть введен в текстовое поле любого размера. Вы можете ввести одну или несколько строк текста на любом из установленных языков.
	Поле ввода/вывода	Поле ввода/вывода может использоваться в отчетах только в качестве поля вывода. В таком поле вы можете выводить значения в следующих форматах: двоичное, дата, дата и время, десятичное, шестнадцатеричное, строка и время.
	Поле даты и времени	Дата и время выводятся в отчете в поле даты и времени. Вы можете выводить системное время или использовать значения тега WinCC flexible, выводимые в поле.
	Графическое поле ввода/вывода	Графическое поле ввода/вывода может использоваться в отчетах только в качестве поля вывода. Графическое поле позволяет выбирать графический элемент из графического списка. При этом вы можете, например, графически отобразить значения тегов состояния.  Пример: Вместо значений 0 и 1, вы можете вывести графическое изображение для открытого и закрытого клапана.
	Символьное поле ввода/вывода	Выпадающий список символьного поля ввода/вывода может использоваться в отчетах только для вывода. Это поле позволяет выбирать текстовый элемент из текстового списка. При этом вы можете, например, отобразить значения тегов в

Символ	Объект	Описание
		текстовом формате. Пример: Вместо значений 0 и 1, для отображения состояния двигателя выводятся тексты "двигатель выключен" или "двигатель включен".
	Окно графики	Этот объект позволяет вставлять в отчет графические изображения, созданные в других программах. Вы можете вставлять графику в следующих форматах: "*.emf", "*.wmf", "*.dib" и "*.bmp." При этом указывается размер окна и свойства графического объекта.

**Замечание**

Некоторые из объектов окна инструментов используются либо с ограниченными функциональными возможностями, либо совсем недоступны. Это зависит от используемого устройства HMI. Недоступные объекты отображаются серыми и не могут быть выбраны оператором.

**10.4.4 Использование объектов отчета****Введение**

В группе "Report objects [Объекты отчета]" находятся специальные объекты для формирования отчетов. Эти объекты предназначены только для использования в отчетах.

**Обзор объектов отчета**

Символ	Объект	Краткое описание
	Номер страницы	Выводит номер страницы отчета. Этот объект вставляется в отчет только один раз, например, в нижнем колонтитуле страницы.
	Печать рецепта	Предназначен для вывода в отчете данных рецепта.
	Печать сообщений	Предназначен для вывода в отчете сообщений.

## 10.5 Формирование отчетов сообщений

### 10.5.1 Отчеты сообщений

#### Введение

В WinCC flexible можно выводить сообщения из буфера сообщений или журнала сообщений в виде отчета.

#### Вывод данных отчета сообщений

Для того чтобы вывести сообщения из буфера сообщений или журнала сообщений в виде отчета, вставьте объект "Print alarm [Печать сообщений]" из окна инструментов в отчет. Выделите объект с тем, чтобы свойства объекта отобразились в окне свойств. В окне свойств определите данные для отчета.

В отчете могут быть выведены следующие данные:

- Текущие сообщения из буфера сообщений
- Сообщения из журнала сообщений

Определите классы сообщений, сообщения которых вы хотите вывести. При этом возможны следующие варианты:

- Error [Сообщения об ошибках]
- Operation [Оперативные сообщения]
- Control [Управление]

Определите последовательность сообщений для вывода.

При этом возможны следующие варианты:

- Первым выводится самое старое сообщение
- Первым выводится самое позднее сообщение

Для вывода сообщений за определенный период свяжите поля "Display beginning [Начало вывода]" и "Display end [Конец вывода]" с тегами. В режиме исполнения в тегах будет храниться дата и время вывода первого и последнего сообщения, сгенерированного в указанный период.

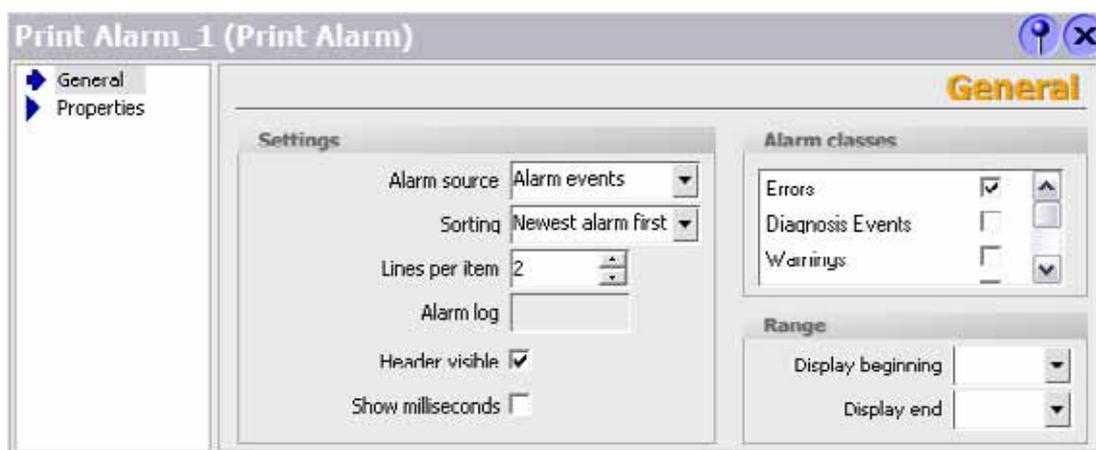
## 10.5.2 Редактирование параметров вывода отчета сообщений

### Введение

Параметры вывода редактируются в окне свойств. Для отображения свойств необходимо чтобы объект "Print alarm [Печать сообщений]" был вставлен в отчет. Должно быть открыто окно свойств.

### Параметры вывода объекта "Print alarm [Печать сообщений]"

Выделите объект в рабочей области. Свойства этого объекта отображаются в окне свойств. Выберите источник данных и укажите параметры вывода в категории "General [Общие]".

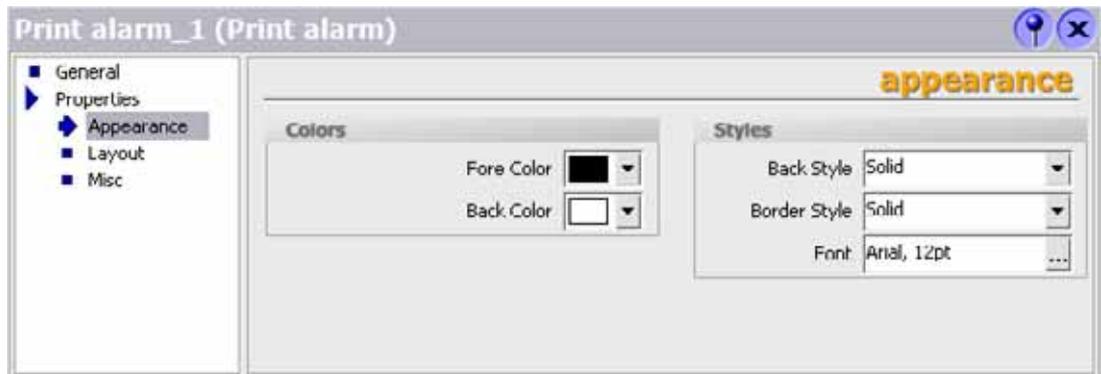


Для вывода сообщений можно определить следующие параметры:

Параметр	Функция	Предварительные условия
"Source for alarms [Источник сообщений]"	Здесь указывается источник сообщений. Возможны следующие варианты: <ul style="list-style-type: none"> <li>События сообщений</li> <li>Журнал/ архив сообщений</li> </ul>	
"Sorting [Сортировка]"	Этот параметр определяет последовательность вывода. Возможны следующие варианты: <ul style="list-style-type: none"> <li>Первым выводится самой старое сообщение</li> <li>Первым выводится самое позднее сообщение</li> </ul>	
"Lines per entry"	Этот параметр указывает	

Параметр	Функция	Предварительные условия
[Количество строк сообщения]"	количество строк, используемых для вывода сообщения. Требуемое количество строк зависит от количества и ширины выбранных для вывода столбцов, а также от используемого шрифта и формата бумаги принтера.	
"Page header visible [Выводить заголовок страницы]"	Определяет, будет ли выводиться шапка таблицы.	
"Alarm log [Журнал сообщений]"	Этим параметром выбирается журнал сообщений для вывода.	В качестве источника данных для вывода "Source for alarms [Источник сообщений]" выбирается журнал сообщений
"Alarm classes [Классы сообщений]"	Используется для выбора классов выводимых сообщений. Возможны следующие варианты: <ul style="list-style-type: none"> <li>• "Сообщения"</li> <li>• "События сообщений"</li> <li>• "Системные сообщения"</li> <li>• "Диагностические события S7"</li> </ul>	
"Display beginning [Начало вывода]"	Используется для выбора первого сообщения при выводе сообщений за указанный период. С помощью списка выбора свяжите поле с тегом. Введите начальное значение для тега в режиме исполнения, например, в поле ввода.	Тег должен иметь тип данных "Date and time [Дата и время]"
"Display end [Конец вывода]"	Используется для выбора последнего сообщения при выводе сообщений за указанный период. С помощью списка выбора свяжите поле с тегом. Введите конечное значение для тега в режиме исполнения, например, в поле ввода.	Тег должен иметь тип данных "Date and time [Дата и время]"

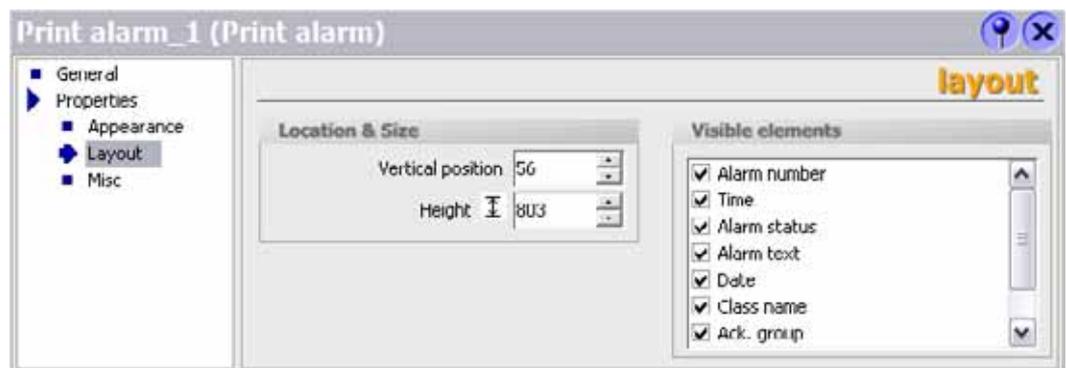
Выберите в категории свойств подкатеорию "Appearance [Вид]". Укажите цвет переднего плана, цвет фона, стиль и шрифт.



Выберите в категории общих свойств подкатегорию "Layout [Макет отчета]". Определите положение и размер объекта "Print alarm [Печать сообщений]". В области "Visible elements [Выводимые элементы]" выберите столбцы, которые будут выводиться в отчете.

В отчете можно вывести следующие столбцы:

- Alarm number [Номер сообщения]
- Time [Время]
- Alarm status [Состояние сообщения]
- Alarm text [Текст сообщения]
- Data [Дата]
- Alarm class [Класс сообщения]
- Acknowledgement group [Группа квитирования]
- Diagnosable [Диагностируемость]
- PLC [ПЛК]



#### Замечание.

Вертикальный размер объекта "Print alarm [Печать сообщений]", указанный при определении параметров отчета не оказывает влияния на объем выводимой информации. Поскольку при выводе отчета могут поступать большие объемы данных, размер объекта "Print alarm [Печать сообщений]", увеличивается динамически так, что все поступающие данные будут выведены в отчете. Если размер страницы превышен, автоматически проставляется разрыв страницы.

## 10.6 Протоколирование данных рецепта

### 10.6.1 Формирование отчетов данных рецепта

#### Введение

В этом разделе описывается формирование отчета или протокола записей рецепта в WinCC flexible.

#### Вывод данных рецепта

Для создания отчета данных рецепта необходимо вставить в отчет объект "Print recipe [Печать рецепта]" из окна инструментов. Выделите объект с тем, чтобы свойства объекта отображались в окне свойств. В окне свойств определите данные для отчета.

Набор данных для вывода в отчете определяется в области "General [Общие]" окна свойств. Выберите записи рецепта для вывода в отчете.

При этом возможны следующие варианты:

- Все записи рецепта
- Записи рецепта из указанного номерами записей диапазона
- Все записи нескольких или всех рецептов
- Записи нескольких или всех рецептов из указанного диапазона

При выборе нескольких рецептов вы можете указать только один диапазон для последовательных рецептов. При этом для выбора используются номера рецептов. Для указания диапазона записей также используются номера записей.

#### Форматирование выводимых данных

В окне свойств, в категории свойств "Properties/ [Свойства/Макет отчета]" определяется, будут ли данные выводиться построчно или в табличной форме. В этой же категории свойств в области "Visible elements [Выводимые элементы]" необходимо выбрать элементы записи, которые будут выводиться в отчете.

Можно вывести следующие элементы записи:

- Recipe number [Номер рецепта]
- Recipe name [Имя рецепта]
- Data record number [Номер записи данных]
- Data record name [Имя записи данных]
- Tag Name [Имя тега]
- "Tag type [Тип данных тега]"
- "Element [Элемент]"

Вывод отчета может быть инициирован событием или наступлением указанного момента времени.

## 10.6.2 Редактирование параметров вывода отчета рецептов

### Введение

Параметры вывода отчета рецептов редактируются в окне свойств. Для отображения свойств в окне необходимо, чтобы объект "Print recipe [Печать рецепта]" был вставлен в отчет. Должно быть открыто окно свойств.

### Параметры вывода объекта "Print recipe [Печать рецепта]".

Выделите объект "Print recipe [Печать рецепта]" в рабочей области. Свойства этого объекта отображаются в окне свойств. Укажите параметры для выбора выводимых в отчете данных рецепта в категории "General [Общие]".



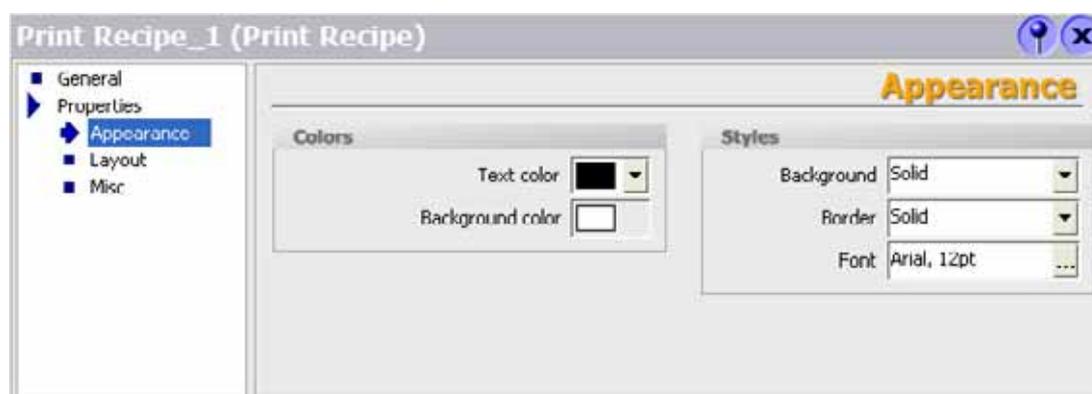
Для выбора выводимых в отчете данных рецепта можно использовать следующие параметры:

Параметр	Функция	Предварительные условия
"Recipe selection [Выбор рецепта]"	Этот параметр позволяет указать критерий выбора выводимых в отчете рецептов. Возможны следующие варианты: "All [Все]" "Name [По имени]" "Number [По номеру]"	
"Recipe name [Имя рецепта]"	Этот параметр позволяет выбрать рецепт по его имени. Если в проекте уже созданы рецепты и определены их свойства, вы можете либо ввести имя рецепта, либо выбрать его из списка объектов.	В поле выбора значения параметра, определяющего критерий выбора выводимых в отчете рецептов "Recipe selection [Выбор рецепта]" выбран вариант "Name [По имени]".
"First recipe"	Этот параметр используется для	В поле выбора значения

Параметр	Функция	Предварительные условия
"[Первый рецепт]"	выбора номера первого из нескольких выводимых в отчете рецептов. Введите в поле фиксированный начальный номер или свяжите это поле с тегом, используя список выбора. При этом в режиме исполнения тегу будет динамически присваиваться начальное значение.	параметра, определяющего критерий выбора выводимых в отчете рецептов "Recipe selection [Выбор рецепта]" выбран вариант "Number [По номеру]".
"Last recipe [Последний рецепт]"	Этот параметр используется для выбора номера последнего из нескольких выводимых в отчете рецептов. Введите в поле фиксированный конечный номер или свяжите это поле с тегом, используя список выбора. При этом в режиме исполнения тегу будет динамически присваиваться конечное значение.	В поле выбора значения параметра, определяющего критерий выбора выводимых в отчете рецептов "Recipe selection [Выбор рецепта]" выбран вариант "Number [По номеру]".
"Data record selection [Выбор записи данных]"	Этот параметр позволяет указать критерий выбора выводимых в отчете записей рецептов. Возможны следующие варианты: "All [Все]" "Name [По имени]" "Number [По номеру]"	
"Data record name [Имя записи данных]"	Параметр используется для выбора записи данных рецепта по ее имени.	В поле выбора значения параметра, определяющего критерий выбора выводимых в отчете записей данных "Data record selection [Выбор записи данных]" выбран вариант "Name [По имени]".
"First data record [Первая запись данных]"	Этот параметр используется для выбора номера первой записи данных из диапазона выводимых в отчете записей рецепта. Введите в поле фиксированный начальный номер или свяжите это поле с тегом, используя список выбора. При этом в режиме исполнения тегу будет динамически присваиваться начальное значение	В поле выбора значения параметра, определяющего критерий выбора выводимых в отчете записей данных "Data record selection [Выбор записи данных]" выбран вариант "Number [По номеру]".
"Last record [Последняя запись данных]"	Этот параметр используется для выбора номера последней записи данных из диапазона выводимых в	В поле выбора значения параметра, определяющего

Параметр	Функция	Предварительные условия
	отчете записей рецепта. Введите в поле фиксированный конечный номер или свяжите это поле с тегом, используя список выбора. При этом в режиме исполнения тегу будет динамически присваиваться конечное значение	критерий выбора выводимых в отчете записей данных "Data record selection [Выбор записи данных]" выбран вариант "Number [По номеру]".

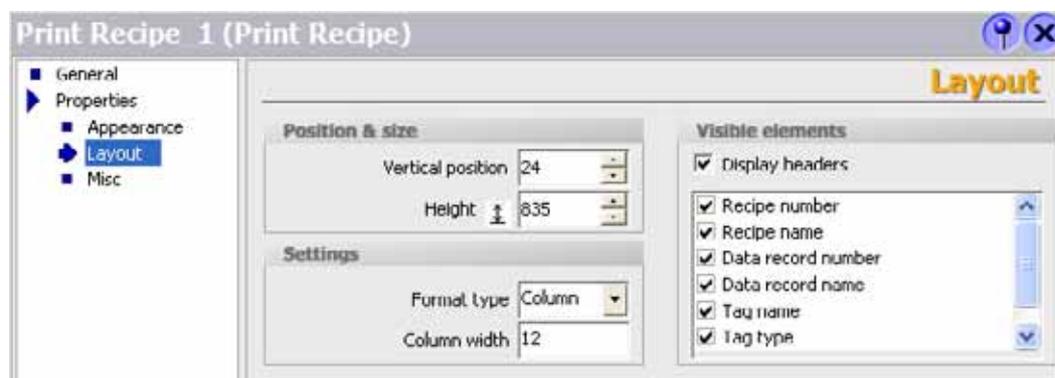
Выберите в категории общих свойств подкатегорию "Appearance [Вид]". Укажите цвет переднего плана выводимого отчета, цвет фона, стиль и шрифт.



Выберите в категории общих свойств подкатегорию "Layout [Макет отчета]". Определите положение и размер объекта "Print recipe [Печать рецепта]" в области "Position and size [Положение и размер]". Изменить положение и размера объекта в рабочей области можно также с помощью мыши.

В области "Settings [Настройка]" выберите табличную и построчную форму вывода. При выборе табличной формы вывода в поле "Column width [Ширина столбца]" укажите ширину столбца таблицы, выбрав соответствующее количество символов. Такую ширину будут иметь все столбцы таблицы.

В области "Visible elements [Выводимые элементы]" выберите выводимые в отчете элементы записи.



**Замечание.**

Вертикальный размер объекта "Print recipe [Печать рецепта]", указанный при определении параметров отчета рецептов, не оказывает влияния на объем выводимой информации. Поскольку при выводе отчета могут поступать большие объемы данных, размер объекта "Print recipe [Печать рецепта]" увеличивается динамически так, что все поступающие данные будут выведены в отчете. Если размер страницы превышен, автоматически проставляется разрыв страницы.

---

## 10.7 Вывод отчета на печать

### Введение

Вывод отчета на печать в WinCC flexible может быть инициирован следующим образом:

Вывод отчета на печать в указанные моменты времени:

- не периодический, в указанные моменты времени,
- периодический, через указанные интервалы времени.

Вывод отчета на печать, инициируемый событием:

- изменением значения тега,
- нажатием сконфигурированной на экране WinCC flexible кнопки,
- переполнением журнала,
- с помощью скрипта в WinCC flexible

### Настройка вывода отчета на печать

Настройка процедуры вывода отчета на печать в указанные моменты времени выполняется с помощью планировщика. Кроме того, вывод отчета на печать может быть запущен системными событиями, которые инициированы планировщиком.

Процедура вывода отчета на печать, инициируемая событием, настраивается непосредственно путем привязки отчета к тегу, конфигурирования соответствующей кнопки на экране или путем привязки к журналу.

---

**Замечание**

Вывод происходит на принтер, указанный как устройство вывода по умолчанию в системе устройства HMI, основанной на Windows.

Для устройств HMI с системами Windows-CE принтер устанавливается с помощью панели управления устройства. Обращение к сетевому принтеру должно происходить по имени. Иными словами, принтер должен быть подключен к сети через DNS сервер. Адресация сетевого принтера с использованием IP адреса на устройствах HMI с системой Windows-CE невозможна.

---

## Администрирование пользователей

### 11.1 Использование системы администрирования пользователей

#### Основные сведения

Система безопасности проекта осуществляет контроль доступа пользователей в к данным и функциям среды исполнения проекта с целью предотвращения несанкционированных действий с их стороны. Разрешение на выполнение операций, имеющих отношение к безопасности системы, дается на этапе разработки проекта и только определенным группам пользователей. Для этого определяются пользователи и группы пользователей, которым назначаются типовые, характерные для данной группы пользователей, права доступа или уровни доступа, называемые также допусками. Объектам проекта также присваиваются требуемые для работы с ними допуски. Например, операторы могут иметь допуск только к указанным функциональным клавишам. Напротив, инженеры-разработчики имеют в режиме исполнения проекта неограниченный доступ к данным и функциям проекта.

#### Определения

Система администрирования пользователей предназначена для централизованного управления пользователями, группами пользователей и правами доступа или допусками пользователей.

Система администрирования осуществляет контроль и управление доступом пользователей к данным и функциям в среде исполнения проекта. С этой целью в системе разработке проекта создаются пользователи и группы пользователей, которым назначаются определенные права доступа. Эти данные передаются на устройство НМІ. Для администрирования пользователей и их паролей в среде исполнения используется "User view [Окно пользователей]".

#### Пример применения

При разработке проекта вы настраиваете защиту от несанкционированного доступа к элементам управления, таким как поля ввода и функциональные клавиши. При этом только определенные пользователи или группы пользователей смогут в режиме исполнения проекта изменять параметры или выполнять настройку и вызывать функции.

### **Осторожно**

Система защиты от несанкционированного доступа не позволяет предотвратить неверные действия оператора. Разработку проекта, ввод проекта в эксплуатацию, сопровождение и обслуживание программного и аппаратного обеспечения системы, а также управление технологическим процессом с использованием проекта WinCC flexible должен выполнять только соответствующим образом подготовленный персонал.

Система защиты от несанкционированного доступа не может использоваться для определения последовательности ежедневных операций, связанных с управлением технологическим процессом и с текущим контролем за их выполнением.

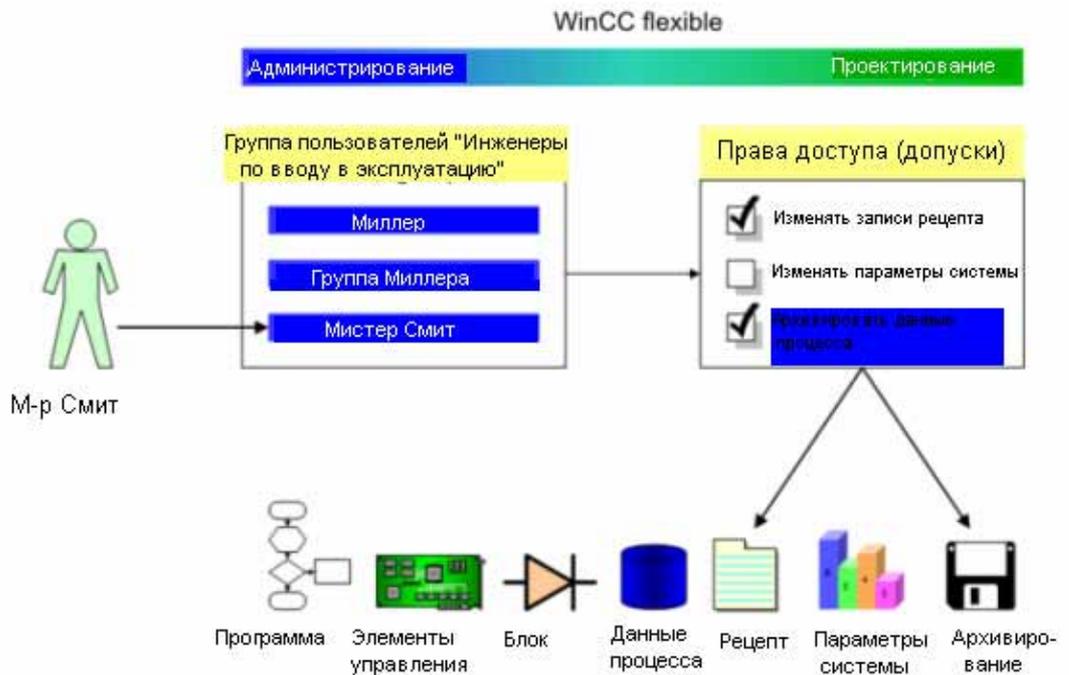
---

## **11.2 Принципы построения системы администрирования пользователей**

### **Введение**

При работе с проектами для управления технологическими процессами в машиностроении необходимо различать пользователей производителя оборудования и пользователей конечного заказчика – операторов установки. Пользователю производителя оборудования, например, м-ру Форману, были предоставлены определенные права доступа при работе с проектом. Однако пользователя Формана не существует у конечного заказчика.

Аналогичные трудности возникают, например, при интеграции различных проектов, связанных с одним технологическим процессом, в один проект. При интеграции таких проектов необходимо в процессе ввода объединенного проекта в эксплуатацию иметь неограниченный доступ к данным отдельных проектов.



Поэтому права доступа в системе администрирования пользователей назначаются не пользователям непосредственно, а группам пользователей. При этом, например, для получения пользователем Форманом определенных прав доступа достаточно включить его в группу "Оператор", которой эти права назначены. То есть права доступа определяются не для каждого конкретного пользователя, а для группы пользователей.

В другой среде, например, у конечного заказчика, существуют другие пользователи. Однако группы пользователей и соответствующие им права доступа, определенные в проекте, остаются неизменными. Необходимо только включить пользователей в соответствующие группы, например, в группу "Оператор".

Таким образом в системе администрирования пользователей процесс управления пользователями отделен от процедуры назначения прав доступа, что делает систему защиты проекта достаточно гибкой.

## 11.3 Элементы пользовательского интерфейса и основная настройка

### 11.3.1 Окно администрирования групп пользователей

#### Введение

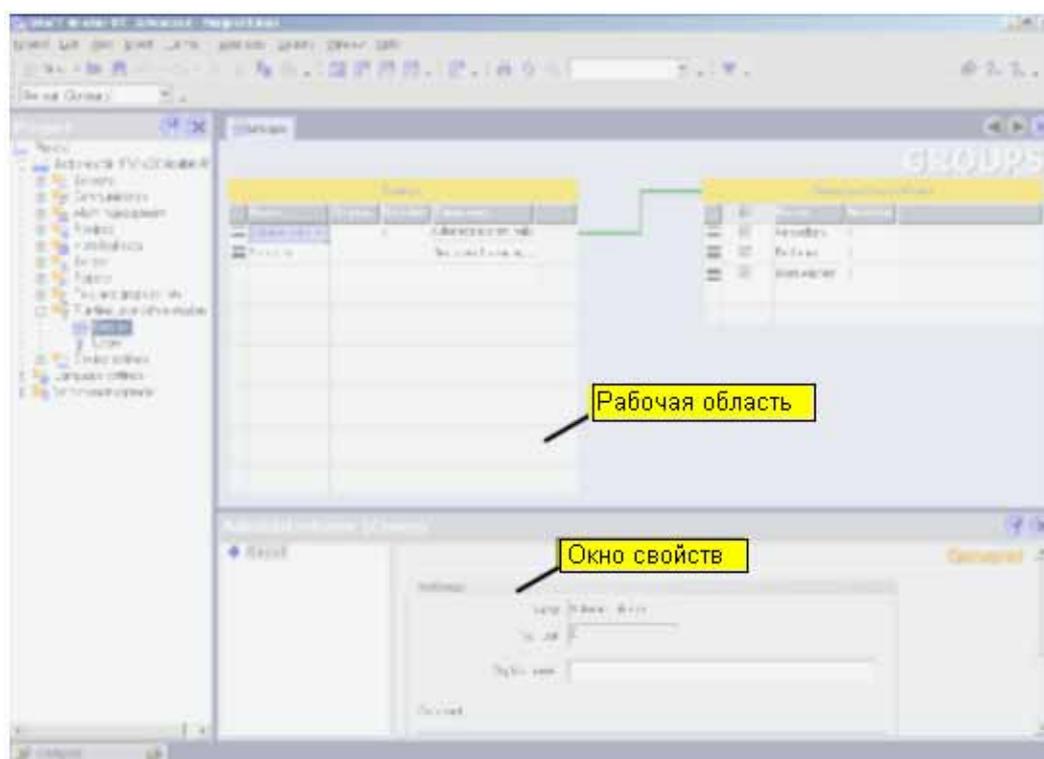
Система администрирования пользователей позволяет контролировать доступ отдельных пользователей и групп пользователей к данным и функциям в режиме исполнения. Администрирование пользователей подразделяется на управление

непосредственно пользователями и управление группами пользователей. В этом разделе описывается администрирование групп пользователей.

### Открытие редактора

Для того чтобы открыть окно администрирования групп пользователей, щелкните дважды на элементе "Groups [Группы]" в окне проекта.

### Структура



### Рабочая область

В рабочей области окна администрирования групп пользователей отображены существующие группы и их права доступа или допуски.

### Окно свойств

В группе свойств "General [Общие]" при выборе пользовательской группы или права доступа вы можете редактировать назначение или комментарий.

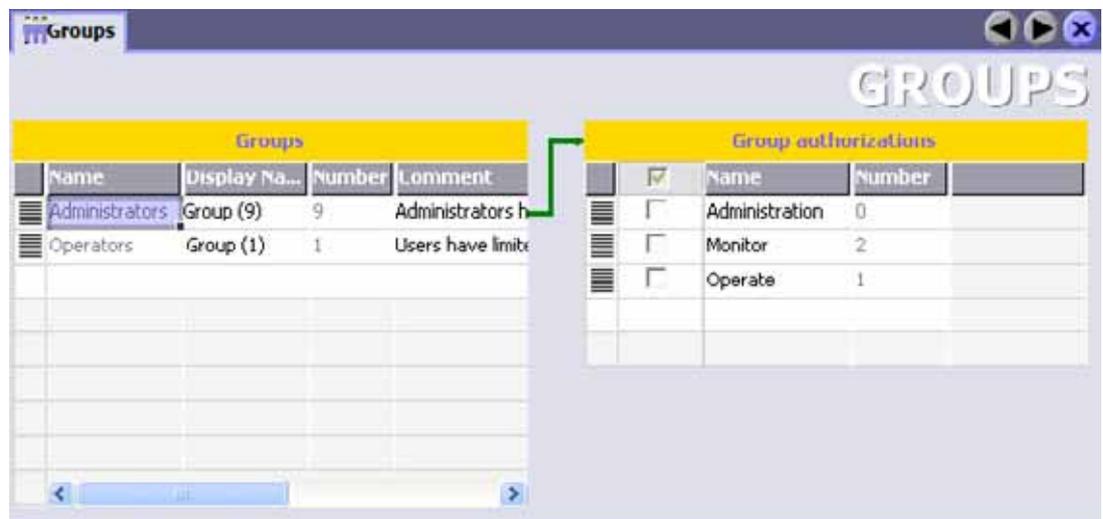
### 11.3.2 Рабочая область окна администрирования групп пользователей

#### Введение

В рабочей области окна администрирования групп пользователей показаны таблицы групп и соответствующих допусков. Здесь группам пользователей назначаются права доступа.

#### Основные сведения

В рабочей области отображаются таблицы "Groups [Группы]" и "Group Authorizations [Права доступа групп пользователей]".



В таблице "Groups [Группы]" показаны существующие группы пользователей. При выборе какой-либо группы в этой таблице, в таблице справа "Group Authorizations [Права доступа групп пользователей]" отображаются допуски, которые были определены для данной пользовательской группы.

Номер группы пользователей и номер соответствующего допуска назначается системой администрирования. Обозначения и описания определяются пользователем.

### 11.3.3 Окно администрирования пользователей

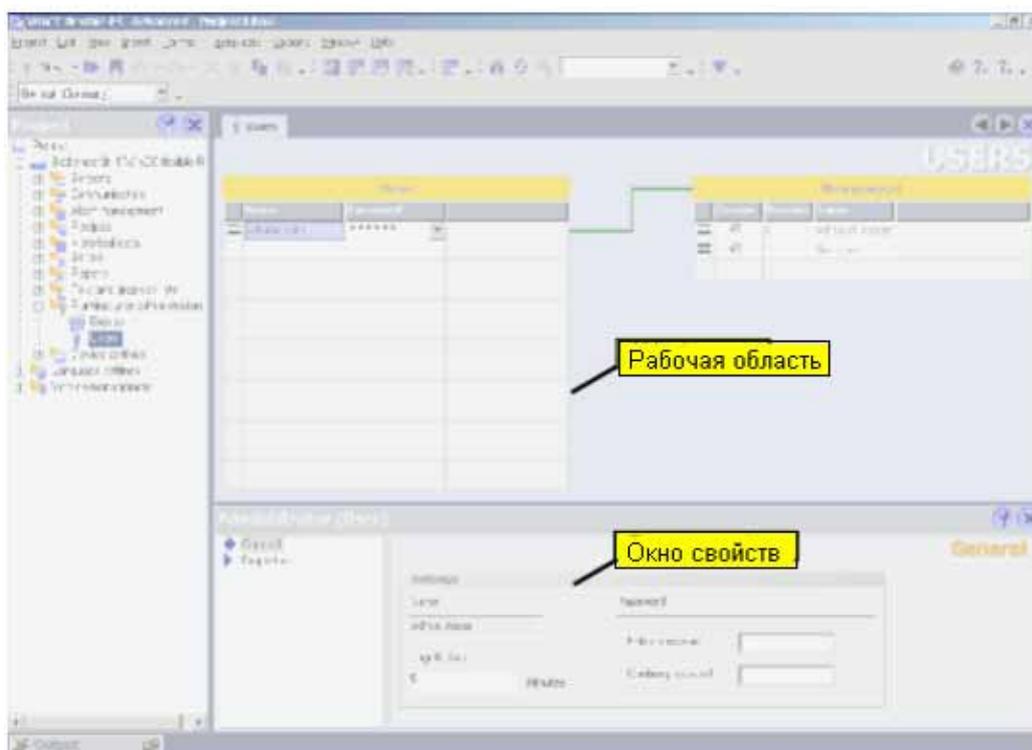
#### Введение

Система администрирования пользователей позволяет контролировать доступ отдельных пользователей и групп пользователей к данным и функциям в режиме исполнения. Администрирование пользователей подразделяется на управление непосредственно пользователями и управление группами пользователей. В этом разделе описывается администрирование пользователей.

#### Открытие редактора

Для того чтобы открыть окно администрирования пользователей, щелкните дважды на элементе "Users [Пользователи]" в окне проекта.

#### Структура



#### Рабочая область

В рабочей области окна "Users [Пользователи]" показаны существующие пользователи и группы, к которым они относятся.

---

#### Замечание

Пользователь может относиться только к одной группе.

---

## Окно свойств

В группе свойств "General [Общие]" при выборе пользователя можно редактировать пароль и время до автоматического окончания сеанса работы в группе.

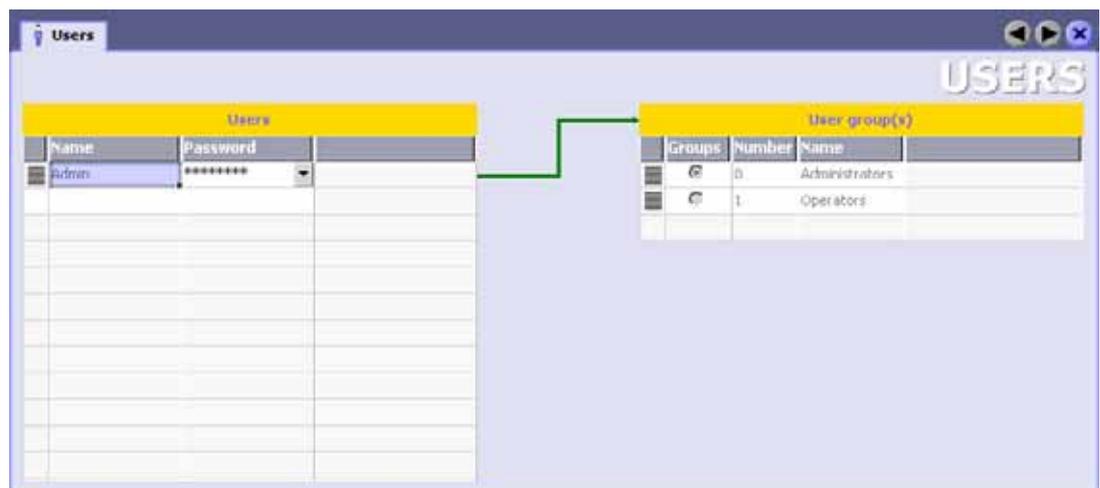
### 11.3.4 Рабочая область окна администрирования пользователей

#### Введение

В рабочей области окна администрирования пользователей представлены две таблицы – таблица пользователей и таблица групп. Здесь вы можете отнести пользователя к определенной группе.

#### Основные сведения

В рабочей области отображаются таблицы "Users [Пользователи]" и "Group of the users [Группы пользователя]".



В таблице "Users [Пользователи]" перечислены существующие пользователи. При выборе пользователя в этой таблице, в таблице "Groups of the user [Группы пользователя]" отображается группа (или группы) пользователей, к которой относится данный пользователь.

## 11.4 Работа с системой администрирования пользователей

### 11.4.1 Пользователи в режиме исполнения

#### Основные сведения

При создании проекта в системе разработки создаются пользователи и группы пользователей и определяются их права доступа. При конфигурировании объектов для них также определяются допуски. После передачи проекта на устройство HMI и запуска проекта все объекты, для которых были определены допуски, защищены от несанкционированного доступа.

#### Окно пользователей

Если при разработке проекта было сконфигурировано окно пользователей, после передачи проекта на устройство HMI и его запуска вы можете использовать это окно для управления пользователями.

---

#### Осторожно

Изменения, внесенные в режиме исполнения в окне пользователей, начинают действовать немедленно. При этом обновления в данных системы разработки проекта не происходит. При передаче данных о пользователях и группах пользователей на устройство HMI все изменения, выполненные в окне пользователей, при соответствующих параметрах передачи и подтверждения пользователем будут утрачены.

На некоторых устройствах HMI окно пользователей не поддерживается. На этих устройствах поддерживается только функции "Log on [Зарегистрироваться в системе]" и "Log off [Завершить работу]": При этом допуск к этим функциям возможен только для пользователя "Administrator [Администратор]". Пользователь "Administrator [Администратор]" относится к единственной группе пользователей "Administrators [Администраторы]".

---

#### Экспорт и импорт данных о пользователях

Данные о пользователях, определенных на устройстве HMI, и их паролях могут быть экспортированы на различные операторские панели и импортированы с различных операторских панелей с помощью системной функции. Это позволяет обеспечить единообразие данных в системах администрирования пользователей на различных устройствах HMI.

---

#### Замечание

Если данные о пользователях были экспортированы из проекта WinCC flexible 2004 в файл экспорта, этот файл может быть затем импортирован в проект WinCC flexible 2005, поскольку WinCC flexible 2005 позволяет использовать данные проектов более ранних версий системы, то есть WinCC flexible 2004.

Если данные были экспортированы в WinCC flexible 2005, этот файл не может быть импортирован в WinCC flexible 2004.

---

## 11.4.2 Защита от неавторизованного доступа

### Введение

Допуск к объекту определяется с целью предотвращения неавторизованного доступа к этому объекту. Доступ к такому объекту могут иметь все зарегистрированные в системе пользователи, которые имеют такой допуск. Если пользователь не имеет допуска к данному объекту, то при попытке обращения к нему автоматически появляется диалоговое окно ввода имени и пароля пользователя.

---

### Замечание

Система администрирования пользователей позволяет использовать несколько системных функций с тем, чтобы данные о пользователях, паролях и группах пользователей можно было редактировать, например, в системе управления.

---



## Использование системных функций и скриптов

### 12.1 Основные сведения

#### 12.1.1 Системные функции и скрипты в режиме исполнения

##### Введение

Для решения общих задач при разработке проекта в WinCC flexible существует ряд готовых системных функций. Использование этих функций для выполнения ряда действий в режиме исполнения не требует специальных навыков программирования.

Для решения более сложных задач среды исполнения проекта можно использовать скрипты. Для доступа к проектным данным в режиме исполнения используется программный интерфейс подготовки скриптов среды исполнения. Создание скриптов среды исполнения может быть выполнено инженерами-проектировщиками, имеющими навыки программирования на языках Visual Basic (VB) и Visual Basic Script (VBS).

##### Использование системных функций

Использование системных функций облегчает работу проектировщика в тех ситуациях, когда операторскому элементу управления необходимо поставить в соответствие одну из функций:

- установка бита в ПЛК
- изменение значения тега;
- запуск процедуры архивирования

Системные функции могут использоваться в списках функций и скриптах.

##### Использование скриптов

Использование скриптов возможно на устройствах HMI, начиная с OP 270/TP 270 и версии системы WinCC flexible Standard. На этих устройствах поддерживается язык программирования VBScript. Использование скриптов, написанных пользователями, позволяет добиться гибкости при разработке проектов. Скрипты следует создавать в

## 12.1 Основные сведения

тех случаях, когда в режиме исполнения проекта необходимы дополнительные функциональные возможности, например:

- преобразование значений

Скрипты можно использовать для преобразования величин, например, из одних единиц измерения в другие.

- автоматизация управления последовательностью технологических операций

С помощью скрипта можно управлять последовательностью выполнения технологических операций путем передачи производственных данных в ПЛК. По значениям, возвращаемым процессом, в скрипте проверяется состояние процесса и, при необходимости, принимаются соответствующие меры.

### Скрипты

Вы можете сохранить код, написанный на языке VB script в виде скрипта. Этот скрипт может использоваться в проекте так же как системная функция. В скрипте вы можете обращаться к тегам проекта и объектной модели среды исполнения WinCC flexible. Кроме того, в скрипте можно использовать все стандартные функции VBS. В скрипте могут вызываться другие скрипты и системные функции.

### Выполнение системных функций и скриптов

В режиме исполнения проекта выполнение системных функций и скриптов инициируется указанным при создании проекта событием (например, щелчком мыши на определенной при проектировании кнопке).

### Уровень рекурсии

Уровень рекурсии или количество вызываемых одна в другой функций в скрипте ограничен размером стека устройства HMI. Неограниченное количество вложенных один в другой вызовов функций может привести в режиме исполнения к возникновению системной ошибки. Поэтому следует ограничивать количество вложений в скрипте.

## 12.1.2 Системные функции

### Введение

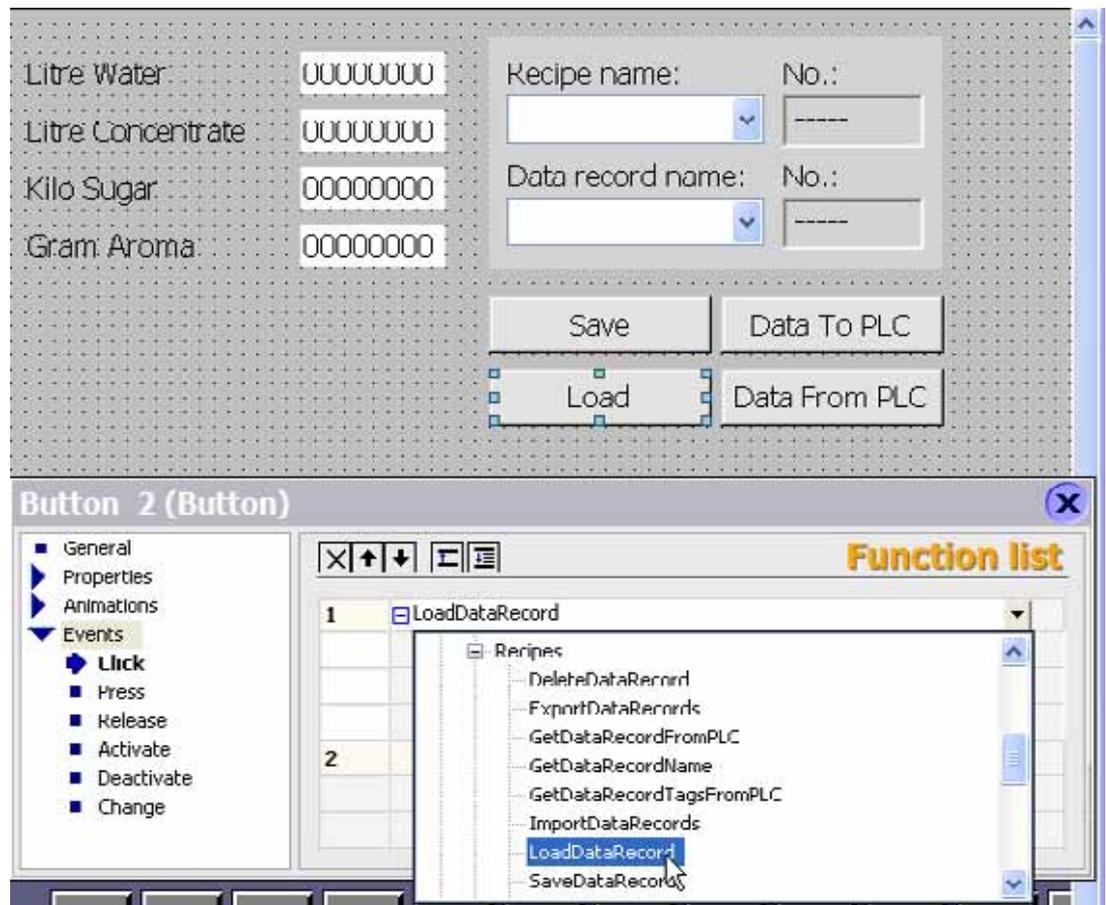
Системные функции представляют собой готовые функции, которые могут использоваться для решения ряда задач, возникающих во время исполнения проекта, даже теми пользователями, которые не имеют навыков программирования. Примерами таких функций могут быть:

- функции вычислений, например, увеличение значения тега на определенную постоянную или переменную величину.
- функции, связанные с журналами, например, запуск процедуры записи в журнал значений процесса.
- функции настройки, например замена ПЛК или установка бита в ПЛК.
- функции генерирования сообщений, например, после смены пользователя.

## Применение

Системные функции могут использоваться в виде списка функций или вызываться в скрипте. Поскольку системные функции представляют собой готовые, ранее созданные функции, их нельзя изменять.

При создании списка функций следует выбирать системные функции из списка выбора, отсортированного по категориям:



Этот же список выбора используется в том случае, если вы хотите использовать системную функцию в скрипте. При этом список выбора вызывается нажатием <Ctrl+Space>.

### Зависимость имени функции от языка проекта

Имя системной функции зависит от установленного языка проекта. При этом функция немедленно распознается планировщиком проекта.

Исключение: При вызове системной функции в скрипте следует использовать имена на английском языке. Английские имена системных функций можно найти в справке по системным функциям.

### Возможные функции

В WinCC flexible можно использовать только те системные функции, которые поддерживаются выбранным устройством HMI. Поэтому только такие функции можно вносить в список функций. Если проект используется на нескольких операторских устройствах, системные функции списка, которые не поддерживаются данным устройством, выделяются цветом.

## 12.1.3 Использование системных функций

### Введение

Выполнение списка функций в режиме исполнения начинается тогда, когда возникает определенное событие. Наступление события может быть вызвано оператором, например, при нажатии функциональной клавиши на операторском устройстве. Событие может быть также сгенерировано системой, например, в том случае, если значение процесса выходит за границы допустимого диапазона.

### Использование системных функций

Вы можете определить системные функции для всех объектов, которые могут определенным образом реагировать на события. Системные функции используются либо в списках функций, либо вызываются в скриптах, оказывая, таким образом, влияние на ход выполнения скрипта.

- Список функций

Системные функции в списке обрабатываются последовательно, начиная с первой и до последней функции. Во избежание задержек, связанных с ожиданием завершения той или иной функции, системные функции, которые исполняются в течение более длительного периода времени, например, операции с файлами, обрабатываются одновременно. Последующая системная функция может начать выполняться даже в том случае, если предыдущая еще не завершена.

Пример формирования списка функций можно найти в разделе "Пример: изменение режима работы на устройстве HMI, на котором отображается информация.

- Скрипт

В скрипте вы можете использовать системные функции в сочетании с кодами других инструкций и условий или условных операторов. Исполнение такого скрипта будет зависеть от состояния системы. Кроме того, значения, возвращаемые системными функциями, могут быть проанализированы в скрипте,

и, в зависимости от результата этого анализа, могут быть вызваны функции, которые, в свою очередь повлияют на ход выполнения скрипта.

## 12.1.4 Скрипты

### Введение

Скрипт представляет собой программу на языке VB script. Законченный скрипт может использоваться в проекте так же, как системная функция. При создании скрипта определяется его тип и передаваемые параметры. Скрипты типа "Function [Функция]" всегда возвращают некоторое значение. Скрипты типа "Sub [Под]" относятся к процедурам и не возвращают никаких значений.

### Свойства скриптов

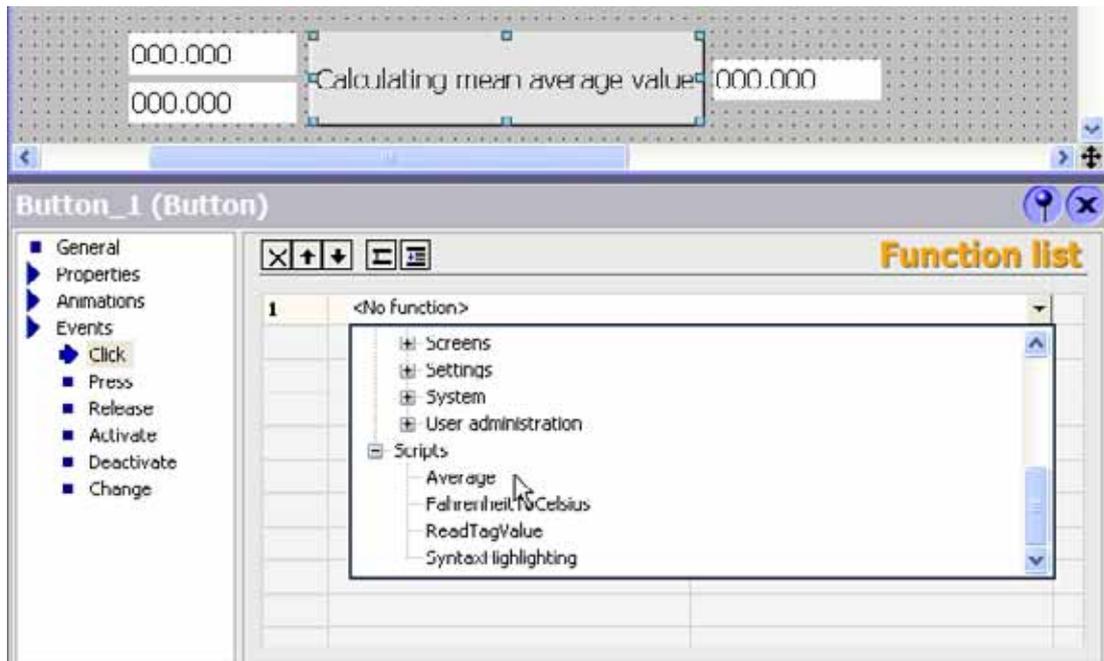
В скрипте могут вызываться другие скрипты и системные функции. Доступ к объектам среды исполнения WinCC flexible осуществляется с помощью объектной модели среды исполнения. При вызове системной функции следует использовать английское имя системной функции. В скриптах можно использовать все возможности языка Microsoft VBScript. Исключаются функции и методы для взаимодействия пользователей, например "MsgBox."

Если вы используете в скрипте системные функции, которые не установлены в системе операторского устройства, будет выведено предупреждающее сообщение. Кроме того, в скрипте соответствующая системная функция будет выделена волнистой голубой линией.

### Организация скриптов

Скрипты хранятся в базе данных проекта. Имеющиеся в проекте скрипты перечислены в окне проекта под элементом Scripts [Скрипты].

Если вы хотите включить скрипт в список функций, его можно найти в списке выбора под элементом Scripts [Скрипты].



## 12.1.5 Использование скриптов

### Основные сведения

Позволяя использовать управляющие элементы языка программирования, скрипты обеспечивают гибкость в управлении процессом.

Использование скриптов позволяет решать конкретные индивидуальные задачи вашего проекта, например:

- Формирование сложного списка функций

Вы можете использовать скрипт в качестве списка функций, вызывая в нем системные функции и другие скрипты.

При этом вы можете написать код скрипта таким образом, что исполнение системных функций и других скриптов, вызываемых в нем, будет происходить в зависимости от соответствующих условий, или повторяться в цикле. Этот скрипт также может быть включен в список функций.

- Программирование новых функций

Скрипты могут использоваться в пределах всего проекта. Вы можете использовать скрипты точно также, как вы используете системные функции. Вы можете определить передаваемые скриптам параметры и использовать возвращаемые скриптами значения параметров. Вы можете использовать скрипты, например, для преобразования значений.

## 12.2 Использование списков функций

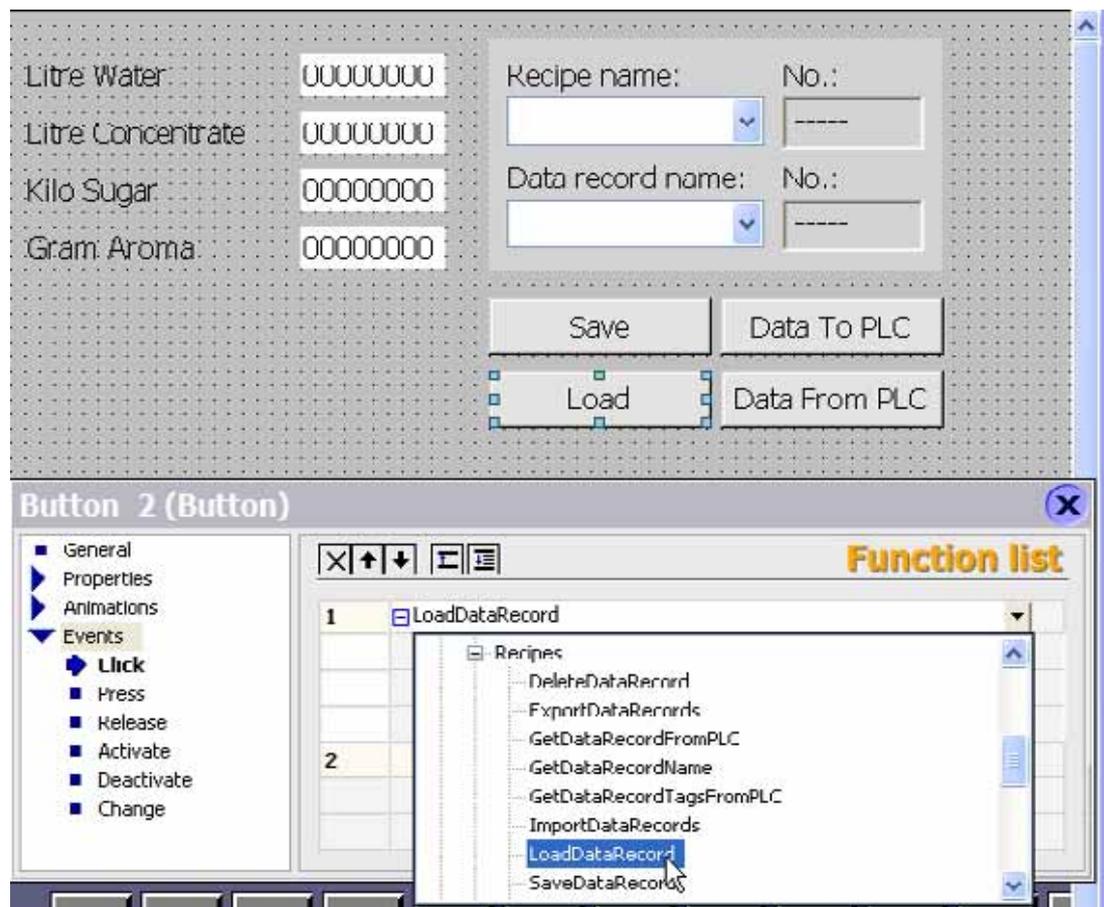
### 12.2.1 Основная информация о списках функций

#### Введение

Список функций позволяет запустить выполнение нескольких системных функций и скриптов при возникновении события, определенного и связанного с данным списком функций при разработке проекта.

#### Принцип использования списка функций

Список функций определяется для события, связанного с объектом проекта, например, экраным объектом или тегом. Возможные события зависят от выбранного объекта и устройства HMI.



События могут происходить только при исполнении проекта. Примерами события являются:

- изменение значения тега,
- нажатие кнопки,
- запуск среды исполнения

Вы можете определить список функций для каждого отдельного события.

---

**Замечание**

Выбор системных функций, включаемых в список, зависит от выбранного устройства HMI.

---

## 12.2.2 Свойства списка функций

### Зависимость списка от устройства HMI

Созданный проект может использоваться на различных устройствах HMI. При замене устройства HMI в проекте, все системные функции и скрипты, не поддерживаемые новым устройством HMI помечаются желтым. Функции, не поддерживаемые устройством, в режиме исполнения не будут выполняться.

### Индикация состояния списка

В процессе проектирования происходит тестирование проектных данных в фоновом режиме. Информация о состоянии, возвращаемая с каждым списком функций, и представленная путем цветовой индикации, показывает состояние соответствующих функций и скриптов.

Состояние показано следующим образом:

- **оранжевый:** в режиме исполнения список функций не выполняется, поскольку хотя бы одной системной функции и одному скрипту списка не были переданы все необходимые параметры.
- **желтый:** в режиме исполнения список функций выполняется. Однако список функций содержит, по крайней мере, одну системную функции или один скрипт, не поддерживаемый устройством HMI (например, в результате изменения типа устройства).

### Выполнение списка системных функций и скриптов

Системные функции и скрипты списка функций обрабатываются в режиме исполнения проекта последовательно, с начала до конца списка. Во избежание задержек, связанных с ожиданием завершения выполнения той или иной функции, системные функции, которые исполняются в течение более длинного периода времени, например, операции с файлами, обрабатываются одновременно. Последующая системная функция может начать выполняться даже в том случае, если выполнение предыдущей еще не завершено.

Для того чтобы избежать необходимости программирования последовательно вызываемых и условных процедур, следует использовать в скриптах циклы, условные операторы и безусловные переходы.

## 12.3 Элементы пользовательского интерфейса и основная настройка

### 12.3.1 Редактор скриптов

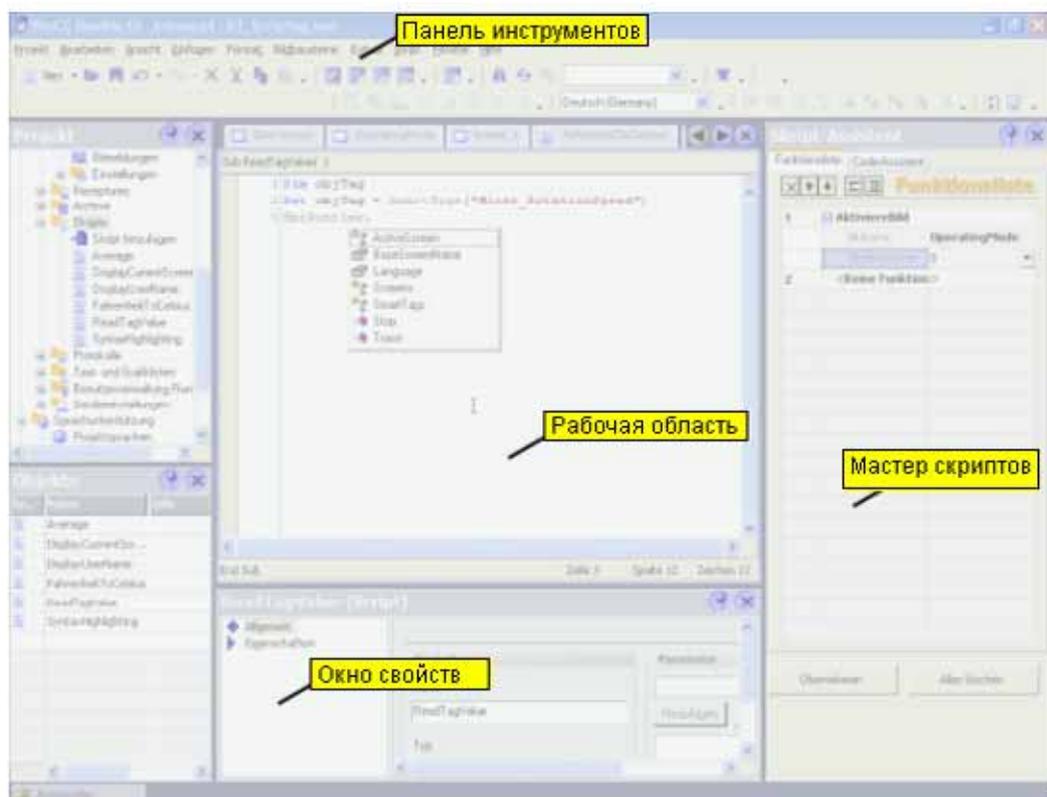
#### Введение

Скрипты создаются и редактируются в редакторе скриптов.

#### Открытие редактора

Редактор скриптов открывается автоматически при создании нового скрипта или открытии существующего.

#### Пользовательский интерфейс редактора



#### Панель меню

Панель меню содержит все команды, необходимые для работы в WinCC flexible. Клавиши быстрого вызова команд показаны рядом с командами меню.

### **Панель инструментов для работы со скриптами**

На панели инструментов редактора скриптов представлены команды для синхронизации объектов и тегов, а также для проверки синтаксиса скриптов.

### **Панель инструментов "Advanced Edit [Расширенные возможности редактирования]"**

На панели инструментов "Advanced Edit [Расширенные возможности редактирования]" представлены команды для работы с закладками, для вставки и удаления кода из скрипта, для создания комментария к коду и для перехода к определенной строке кода.

### **Панель инструментов "IntelliSense [Интеллектуальные средства программирования]"**

На панели инструментов "IntelliSense [Интеллектуальные средства программирования]" представлены команды для отображения списков выбора, например, объектов объектной модели, системных функций или констант VBS.

### **Рабочая область**

Скрипты создаются и редактируются в рабочей области. Создание скриптов упрощается благодаря проверке синтаксиса и интеллектуальным средствам программирования.

### **Окно свойств**

Свойства скриптов определяются в окне свойств. Здесь, в частности, определяется, является ли скрипт процедурой или функцией. Более того, здесь вы можете объявить параметры скрипта.

### **"Мастер скриптов"**

В Мастере скриптов ("Script Wizard") могут быть определены параметры системных функций и скриптов точно также как и в списке функций.

Заполненные параметрами функции и скрипты могут быть перемещены из Мастера скриптов в активный скрипт. При этом вам необходимо выполнить определение параметров только один раз.

Если системные функции или скрипты уже определены разработчиком и связаны с событием, они могут быть скопированы в Мастер скриптов с помощью операций копирования и вставки. Определение параметров системных функций в Мастере скриптов возможно только для тех функций, использование которых допустимо в скрипте. При копировании в Мастер скриптов с помощью операции копирования и вставки системных функций, которые не могут использоваться в скрипте, они будут помечены.

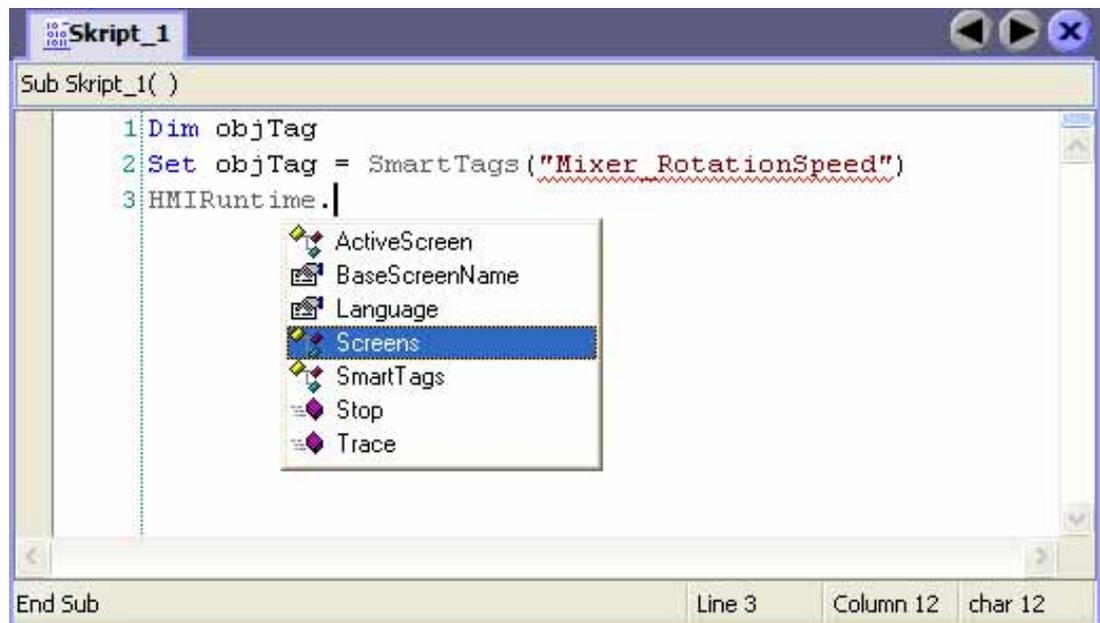
## 12.3.2 Возможности редактора скриптов

### Введение

Процесс программирования с использованием редактора скриптов значительно облегчается благодаря использованию таких средств как интеллектуальные средства программирования, функции проверки синтаксиса и других. Например, ссылки на теги могут быть созданы с помощью функции перетаскивания.

### IntelliSense [Интеллектуальные средства программирования]

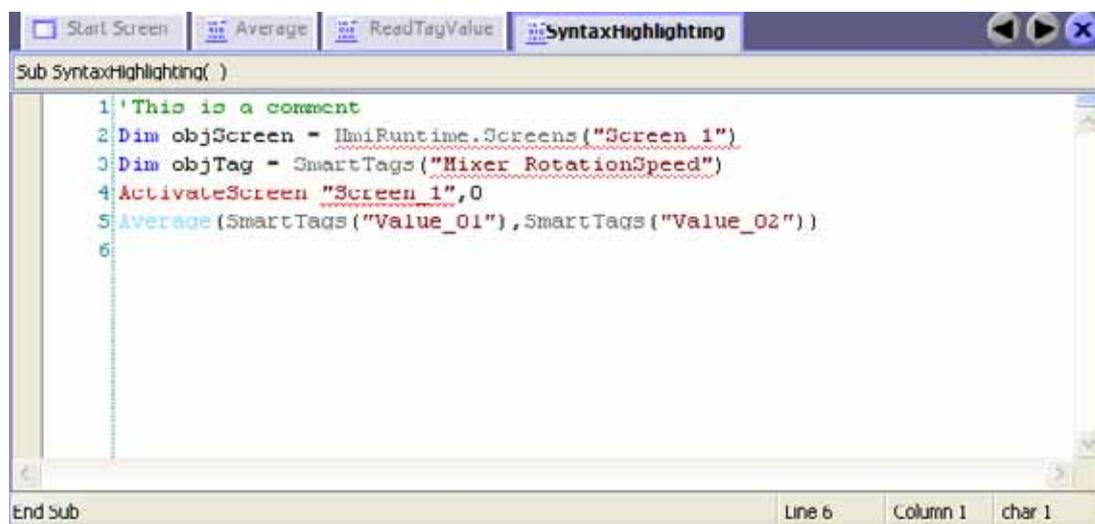
Интеллектуальные средства программирования облегчают процедуру создания скрипта при обращении к объектам, функциям и свойствам моделей объектов VBS:



Функции и свойства данного объекта могут быть выбраны из списка выбора.

## Проверка синтаксиса

В редакторе скриптов ключевые или зарезервированные слова выделяются различными цветами. Объекты, распознаваемые редактором скриптов, выделяются жирным шрифтом. Незвестные слова подчеркиваются красной волнистой линией:



В приведенной ниже таблице показаны цвета, принятые для наиболее важных ключевых слов.

Цвет	Объект, выделяемый этим цветом	Пример
Синий	Ключевое/зарезервированное слово (VBS)	Dim, If, Then
Серый	Ключевое слово (модель объекта)	HMI runtime
Голубой	Скрипт	Fahrenheit to Celsius
Коричневый	Системная функция	IncreaseValue
Красный	Тег	Tag_1
Зеленый	Комментарий	'This is a comment

## Синхронизация объектов

При открытии скрипта экземпляры объектов (например, теги) автоматически синхронизируются с проектными данными. Если, например, тег был переименован в редакторе тегов, то это изменение также должно быть выполнено в скрипте. После переименования тега и открытия скрипта, в котором содержится данный тег, переименованный тег будет подчеркнут волнистой синей линией. Теперь можно выполнить синхронизацию в редакторе скриптов вручную.

## Список объектов

Список объектов можно вызвать с помощью клавиш <Alt+Right>. В этом списке, в зависимости от типа объекта, будут отображены все доступные в проекте объекты этого типа. Списки объектов могут вызываться, например, во время определения

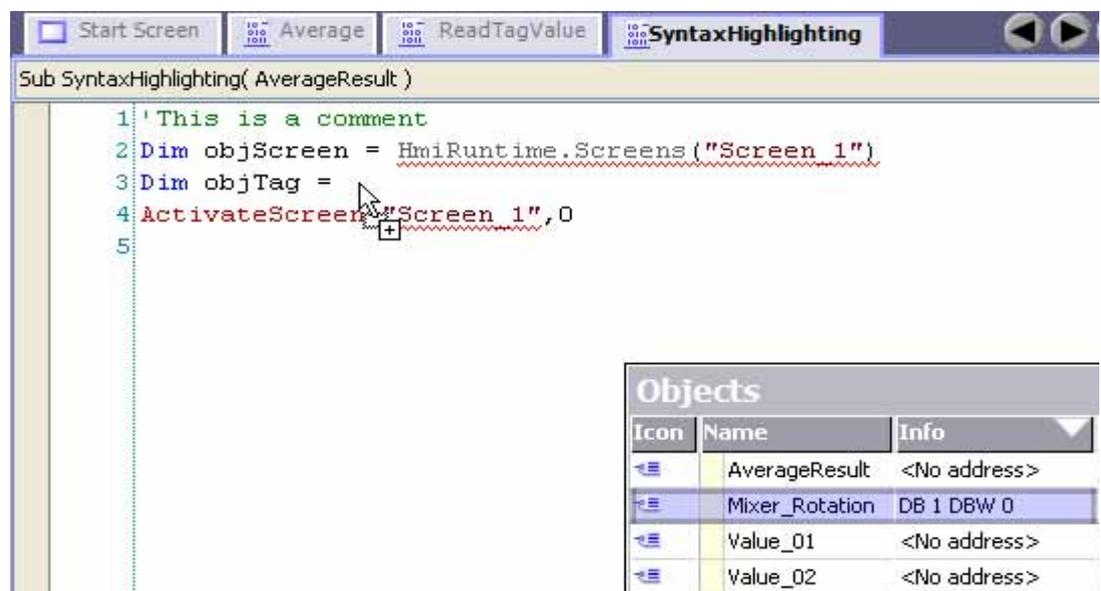
параметров, или при необходимости распечатать перечень объектов определенного типа.

Пример: вы хотите вставить ссылку на некоторый экран процесса, используя при этом список экранов проекта. Введите в редакторе скриптов `HmiRuntime.Screens` и затем вызовите список объектов с помощью клавиш `<Alt+Right>`. В списке будут перечислены все доступные в проекте экраны процесса:

Выберите требуемый экран в списке и вставьте его в скрипт, нажав `<Return>`.

## Функция перетаскивания

Если в скрипте необходим тег, его можно перетащить из окна объектов.

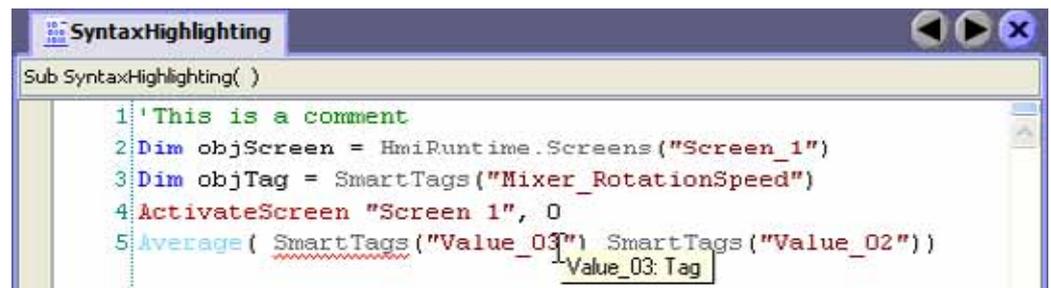


## Функции справки

В процессе программирования вам будут автоматически предоставлены краткие описания необходимых параметров для функций объектов и системных функций. Кроме того, в редакторе скриптов вы можете пользоваться следующими функциями справки:

- Всплывающая подсказка

Неизвестные или неверно написанные ключевые слова будут подчеркиваться волнистой линией. При перемещении курсора мыши с ключевого слова подсказка исчезает:



Для известных ключевых слов всплывающая подсказка показывает тип ключевого слова.

- Информация о параметрах (ParametrInfo)

Этот вид справки предоставляет информацию о синтаксисе и параметрах системной функции или стандартной функции VBS.

- Контекстно зависимая справка

В контекстно - зависимой справке содержится информация о системных функциях, элементах языка VBS, объектах и т.д.

Если необходима информация об объекте, методе или свойстве, поместите указатель мыши к соответствующему ключевому слову и нажмите клавишу <F1>. В результате этого действия вы сможете получить требуемую информацию на соответствующей странице интерактивной справочной системы.

## 12.4 Создание скриптов

### 12.4.1 Обращение к тегам

#### Введение

В скрипте вы можете обращаться к внутренним и внешним тегам, определенным в проекте. При этом в режиме исполнения значение тега, используемого в скрипте может быть считано по соответствующему адресу или изменено. Кроме того, в скрипте вы можете определить локальные теги, которые будут выполнять функции счетчика или буфера памяти.

#### Теги проекта

Если имя тега в проекте удовлетворяет правилам именования переменных в VBS, этот тег можно использовать непосредственно в скрипте:

```
"VBS_Example_03
```

```
If BeltDriveOilTemperature > 100 Then [instruction]
```

Если имя тега не удовлетворяет правилам, принятым в VBS, к нему нужно обращаться через список "Smart tags". В следующем примере имя тега содержит символ &, недопустимый в имени в соответствии с правилами именования, принятыми в VBS:

```
"VBS_Example_04
```

```
Dim objTag
```

```
SetobjTag = SmartTags("Test&Trial")
```

Правила именования, принятые в VBS можно найти в информационной справочной системе, в разделе, посвященном VBS.

## Локальные теги

Локальные теги могут быть определены в скрипте с помощью оператора Dim. Локальные теги могут использоваться только в пределах данного скрипта. Поэтому в редакторе тегов локальные теги не отображаются.

Например, локальный тег может использоваться в скрипте в качестве счетчика в операторе For.

```
'VBS_Example_05Dim intCountFor intCount = 1 To 10[Instruction]Next
```

---

### Замечание

Если в операторе "For" необходим тег, вы должны использовать локальный тег. Теги проекта использовать в операторе "For" не допускается.

---

## 12.4.2 Вызов скриптов и системных функций в скрипте

### Основные сведения

В скрипте можно вызывать системные функции и другие скрипты.

Вызов в скрипте системной функции или скрипта без возвращаемого значения (без выходного параметра) (тип "Sub") выполняется следующим образом

```
<Function name> [Parameter1], [Parameter2], [...]
```

Вызов системной функции или скрипта с возвращаемым значением (выходным параметром) (тип "Function") выполняется путем присвоения этой функции некоторому выражению:

```
<Expression> = <Function name> ([Parameter1, Parameter2,... [Parameter N])
```

Если вы не собираетесь использовать возвращаемое скриптом значение, используйте вариант вызова системной функции или скрипта без возвращаемого значения.

### Особенности вызова системных функций

Системные функции и скрипты могут вставляться в скрипт в Мастере скриптов ("Script Wizard"). При этом системные функции отображаются в Мастере скриптов на установленном в данный момент языке проекта.

При вызове системной функции в скрипте следует всегда использовать английское имя системной функции:

```
SetValue Tag1, 64
```

Английские имена системных функций можно найти в справке по системным функциям в разделе "Syntax [Систаксис]". Установленный в проекте язык во внимание не принимается.

При передаче параметров системным функциям действуют следующие правила:

- Передача в качестве параметров констант.

Если в качестве параметра используется константа, параметр должен иметь один из трех следующих типов данных: целое, двойное слово или строка. Возможные

константы показаны в списке выбора при определении параметров. К константам применимы обычные правила именования VBS.

- Передача тегов в качестве параметров.

Вне зависимости от написания теги всегда передаются по принципу "обращение по ссылке (Call by reference)". Если передаваемый тег удовлетворяет правилам именования в VBS, имя тега может быть указано в качестве параметра без использования ключевого слова smart tags:

```
SetValue Tag1, 64
```

или

```
SetValue SmartTags ("Tag1"), 64
```

- Ссылка на объект, например, экран процесса, соединене или журнал

Объект, который передается в качестве параметра, заключается в кавычки:

```
ActivateScreen "MainScreen", 0
```

### Особенности вызова скриптов

При вызове скрипта параметры передаются по принципу "обращение по ссылке". При передаче тега в качестве параметра, например, присвоение этому тегу некоторого значения в скрипте вызовет немедленное изменение значения тега.

### Зависимость кода скрипта от операторского устройства

Код скрипта зависит от выбранного операторского устройства. Если в скрипте используются системные функции, не поддерживаемые выбранным операторским устройством в окне вывода появляется сообщение об ошибке.

## 12.4.3 Обращение к объектам

### Введение

В скрипте можно использовать объекты модели среды исполнения с соответствующими свойствами и методами.

В режиме исполнения можно читать и редактировать свойства объектов.

### Обращение к объектам

Обращаться к объектам в скрипте следует через соответствующий список. Укажите имя объекта или номер его позиции в списке.

Обратиться к первому объекту экрана "Main screen [Главный экран]" можно с помощью следующих операторов:

```
"VBS_Example_01
```

```
Dim objObject
```

```
'Change to Screen "MainScreen"
```

```
HMIRuntime.BaseScreenName = "MainScreen"
```

```
Set objObject = HMIRuntime.Screens(1).ScreenItems(1)
```

Обращение к объекту по имени и изменение его свойств можно выполнить с помощью следующих ниже операторов: для этого при определении объекта на экране необходимо указать его имя.

```
'VBS_Example_02
```

```
Dim objCircle
```

```
HMIRuntime.BaseScreenName = "MainScreen"
```

```
Set objCircle = HMIRuntime.Screens(1).ScreenItems("Circle_01")
```

```
objCircle.BackColor = vbGreen
```

## 12.4.4 Синхронизация тегов и объектов

### Введение

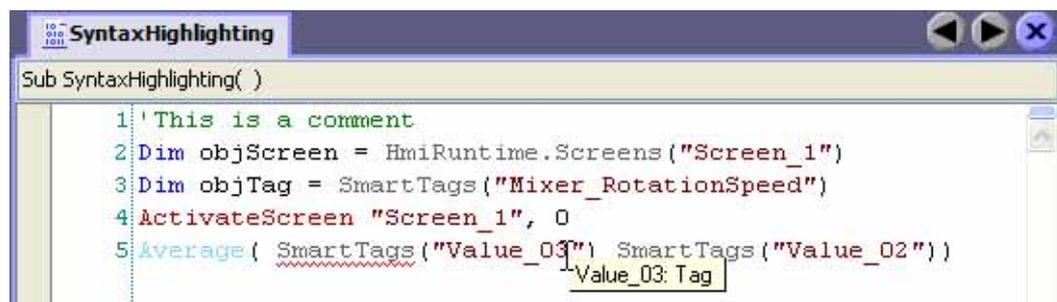
При изменении имени объекта в WinCC flexible это изменение действует в пределах всего проекта. Такие изменения показаны в скрипте как "Synchronizing [Синхронизация]".

### Пример применения

В редакторе тегов определен тег "Oil Temperature [Температура масла]", который будет использоваться в скрипте. Переименуйте этот тег, задав в редакторе тегов новое имя - "OilTemperatureMotor1 [Температура масла в двигателе 1]".

- Во время переименования тега скрипт был открыт:

Старое имя тега будет в скрипте подчеркнуто волнистой синей линией. При перемещении указателя мыши к имени тега появляется подсказка. Щелкните на кнопке "Synchronize [Синхронизировать]". Тег в скрипте будет переименован:



```
Sub SyntaxHighlighting( )  
1 'This is a comment  
2 Dim objScreen = HmiRuntime.Screens("Screen_1")  
3 Dim objTag = SmartTags("Mixer_RotationSpeed")  
4 ActivateScreen "Screen_1", 0  
5 Average( SmartTags("Value_03") SmartTags("Value_02"))  
           Value_03: Tag
```

- Во время переименования скрипт был закрыт

При повторном открытии скрипта тег будет автоматически синхронизирован.

## 12.5 Отладка

### 12.5.1 Отладка скриптов

#### Введение

Процедура отладки позволяет протестировать скрипты в среде исполнения проекта на предмет отыскания ошибок программирования. Например, при отладке можно проверить, верные ли значения были присвоены тегам и верно ли реализованы условные переходы.

Для отладки скриптов используйте отдельно приобретаемый "Microsoft Script Debugger [Отладчик скриптов Microsoft]" или "Microsoft Script [Редактор скриптов Microsoft]", поставляемый с Microsoft Office XP.

#### Типы ошибок

При отладке могут быть выявлены ошибки следующих типов:

- Ошибка среды исполнения

Ошибка среды исполнения возникает при попытке выполнить недопустимую или неверную инструкцию, например оператор, в котром не определен тег.

Для того чтобы иметь возможность фиксировать ошибки среды исполнения, вы можете использовать в скрипте инструкцию "On error resume next [При возникновении ошибки перейти к следующему оператору]". Эта инструкция вызывает после возникновения ошибки среды исполнения выполнение последующих инструкций. Код ошибки можно проверить с помощью объекта ошибки в следующей строке. Для того чтобы остановить обработку ошибок среды исполнения в скрипте, используйте инструкцию "On error go to 0 [при возникновении ошибки перейти к 0]". Дополнительную информацию об обработке ошибок можно найти в информационной справочной системе в разделе, посвященном Microsoft VBS.

- Логическая ошибка

Логическая ошибка возникает в том случае, когда ожидаемое событие не происходит, например, потому что неверное было проверено условие. Для устранения логической ошибки необходимо проследить выполнение скрипта шаг за шагом с тем чтобы обнаружить, какая из его частей не работает.

### 12.5.2 Интеграция отладчика в систему

#### Установка отладчика скриптов в проекте WinCC flexible

Для поиска ошибок в скриптах проектов WinCC flexible необходимо установить отладчик скриптов.

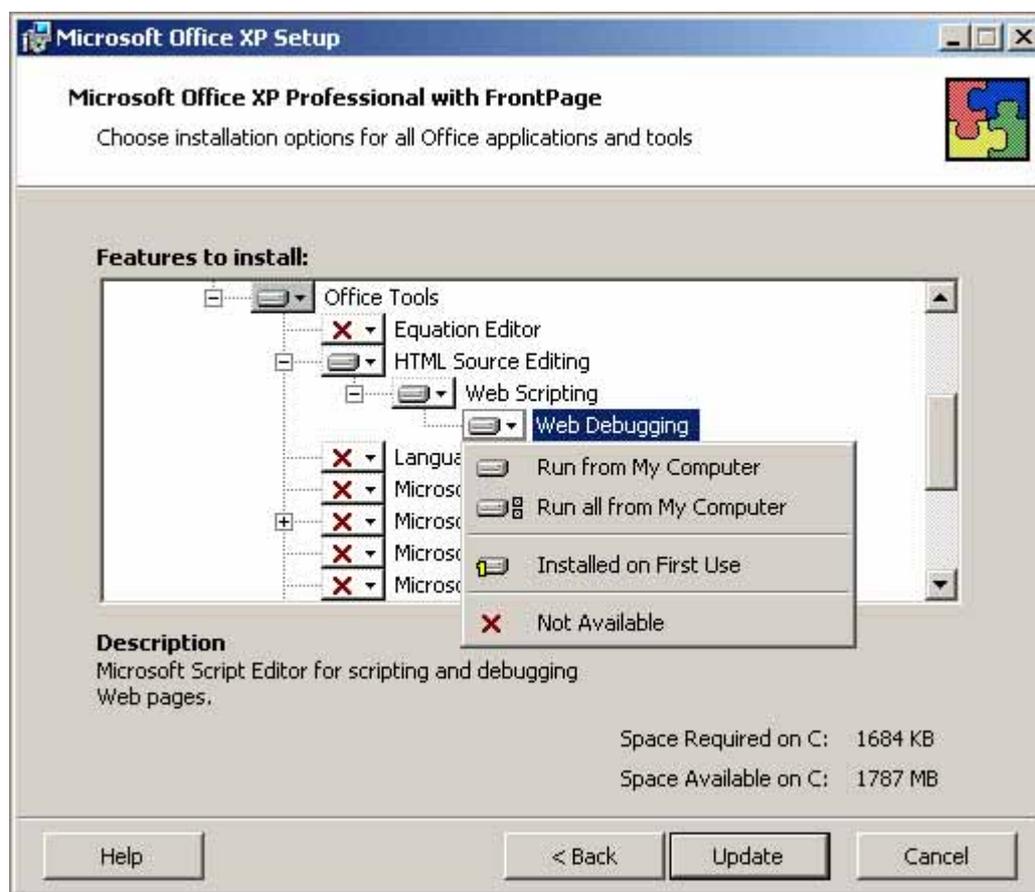
Были выпущены и протестированы следующие отладчики скриптов:

- Microsoft Script Editor [Редактор скриптов Microsoft], входящий в пакет Office XP
- Microsoft Script Debugger [Отладчик скриптов Microsoft]

Установленный отладчик будет запускаться автоматически при возникновении ошибки среды исполнения в скрипте или вручную с помощью команды "Start runtime system with script debugger [Запустить среду исполнения с отладчиком скриптов]".

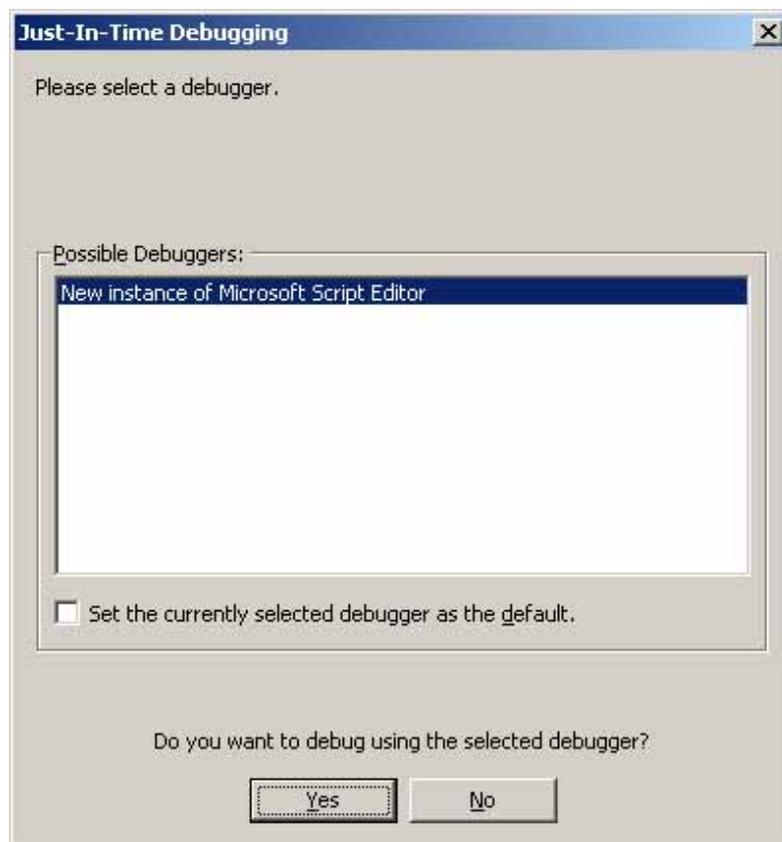
### Microsoft Script Editor [Редактор скриптов Microsoft]

Компонент "Microsoft Script Editor [Редактор скриптов Microsoft]" пакета Microsoft Office XP содержит отладчик скриптов. Если при установке Microsoft Office были использованы параметры настройки по умолчанию, компонент "Microsoft Script Editor [Редактор скриптов Microsoft]" будет установлен при первом использовании, как указано параметром ("Installed on First Use"). Если вы хотите установить этот компонент явно, вы должны указать эту опцию при установке Microsoft Office. При выборе компонентов в окне выбора щелкните на "Web Debugging" и выберите опцию "Run from My Computer [Запускать с моего компьютера]".

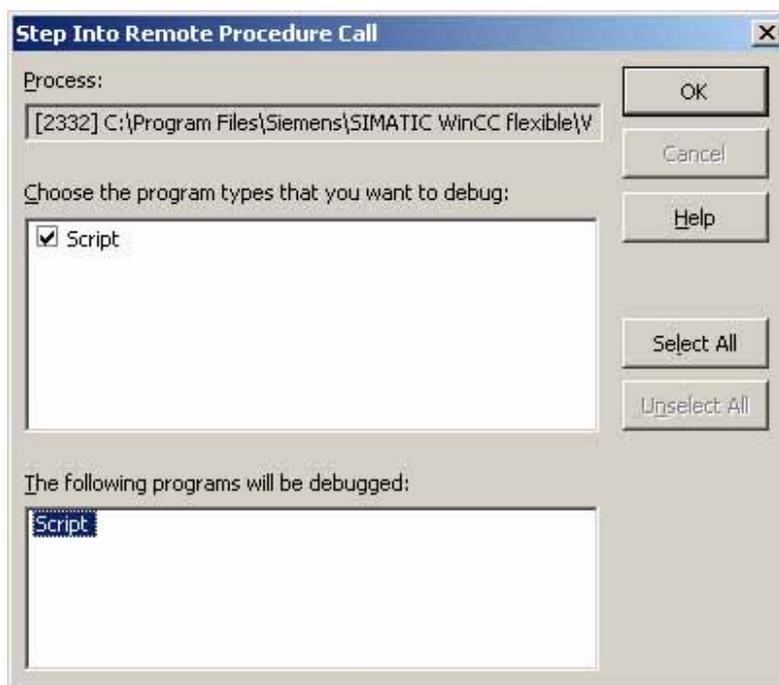


Если проект был запущен в WinCC flexible с опцией "Start runtime system with script debugger [Запустить среду исполнения с отладчиком скриптов]" при запуске исполнения первого скрипта откроется диалоговое окно со списком возможных отладчиков скриптов.

В списке могут появиться другие отладчики скриптов, такие как "Microsoft Visual Interdev" или "Microsoft Visual Studio .NET". Выберите "Microsoft Script Editor [Редактор скриптов Microsoft]" и подтвердите выбор нажатием кнопки "Yes [Да]".



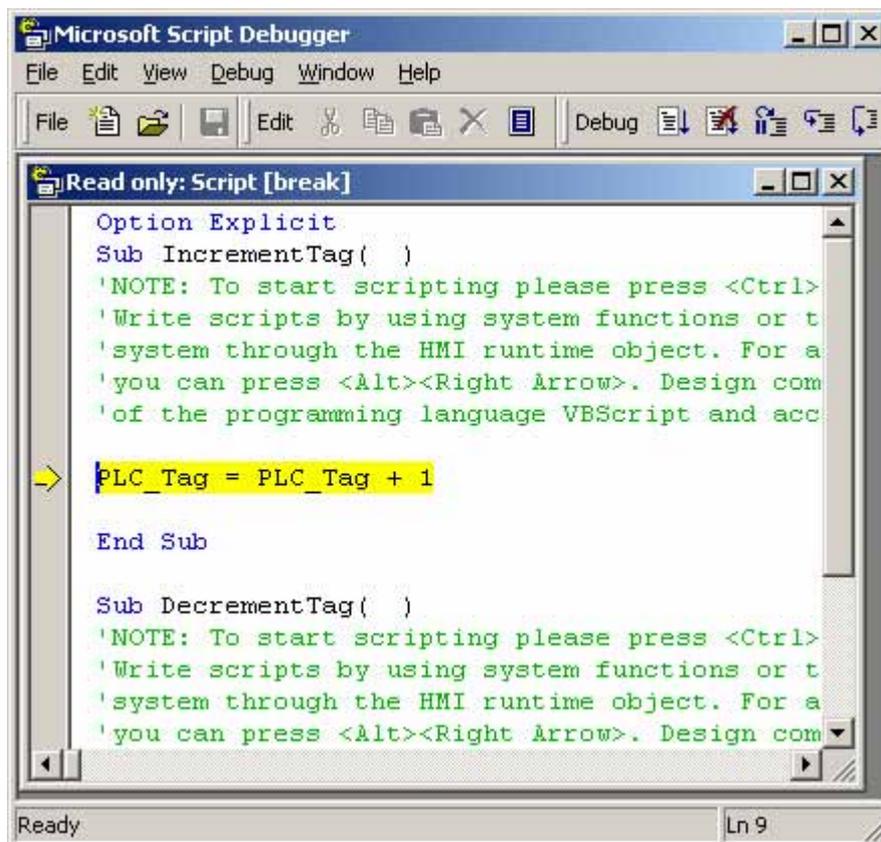
Установите флажок опции "Script [Скрипт]" в диалоговом окне "Step Into Remote Procedure Call [Перейти к удаленному вызову процедур]" и подтвердите выбор, нажав "OK".



Будет запущен "Microsoft Script Editor [Редактор скриптов Microsoft]" и выполнение будет остановлено на первой строке скрипта.

### Microsoft Script Debugger [Отладчик скриптов Microsoft]

Если нет ни одного доступного отладчика, вы можете загрузить бесплатный отладчик "Microsoft Script Debugger" (scd10en.exe) с сайта компании Microsoft ([www.microsoft.com](http://www.microsoft.com)). Он будет автоматически запущен в WinCC flexible сразу же после установки.



### Замечание

"Microsoft Script Debugger [Отладчик скриптов Microsoft]" не поддерживается системой, если на вашем компьютере установлена какая-либо другая система отладки скриптов!

### Отладчик скриптов не стартует при запуске среды исполнения

Если вы установили отладчик скриптов, но он не стартует при использовании опции "Start Runtime with Script Debugger [Запустить среду исполнения с отладчиком скриптов]", проверьте в системном реестре следующие записи:

1. Следующая запись должна быть представлена в реестре для активизации "Just in Time Debugging [Отладка при исполнении]":  
[HKEY\_CURRENT\_USER\Software\Microsoft\Windows Script\Settings]  
"JITDebug"=dword:00000001

Любые внесенные изменения начинают действовать только после перезагрузки компьютера.

2. Для отключения отладки скриптов с использованием Internet Explorer должна быть представлена следующая запись:  
[HKEY\_CURRENT\_USER\Software\Microsoft\Internet Explorer\Main]  
"Disable Script Debugger" = "yes"

## 12.6 Выполнение функций в среде исполнения проекта

### 12.6.1 Выполнение списка функций при исполнении проекта

#### Основные сведения

Функции списка выполняются в режиме исполнения проекта с начала до конца списка. Различают последовательное и параллельное выполнение, при котором отсутствуют периоды ожидания окончания выполнения некоторых функций. Система устанавливает это отличие путем сравнения различных периодов времени выполнения системных функций. Скрипты всегда обрабатываются последовательно, независимо от среды исполнения. Если системная функция возвращает состояние ошибки, выполнение списка функций прекращается.

#### Последовательное выполнение

При последовательном выполнении системные функции списка функций выполняются одна за другой. До того как начнется выполнение следующей функции, должно закончиться выполнение предыдущей.

#### Параллельное выполнение

Системные функции, выполняющие операции с файлами, такие как сохранение и чтение, выполняются дольше, чем функции, которые, например, присваивают значение тегу.

Поэтому, системные функции с более длительным временем исполнения выполняются параллельно. Пока выполняется процедура записи данных на носитель, начинается выполнение следующей системной функции. Благодаря параллельному выполнению системных функций не возникает задержек по причине ожидания окончания выполнения функций.

### 12.6.2 Обработка скриптов при исполнении проекта

#### Основные сведения

В режиме исполнения проекта в данный момент времени может выполняться только один скрипт. Если редактирования ожидают несколько скриптов, формируется очередь скриптов и скрипты выполняются один за другим.

---

#### Замечание

Цикл в скрипте блокирует выполнение других скриптов даже в том случае, если запуск выполнения скриптов происходит параллельно.

---

WinCC flexible позволяет использовать максимально количество вложений, равное восьми. Обратите внимание на то, что проверки количества вложений не выполняется.

---

**Замечание**

Если создается скрипт, связанный с событием "Shutdown [Завершение работы]", в нем можно использовать только те функции, которые в вызове системной функции "Shutdown" определены как настраиваемые объекты.

Убедитесь в том, что выполнение этого скрипта не будет препятствовать завершению режима исполнения.

---

### 12.6.3 Передача и возврат значений параметров скриптов

#### Передача значения скрипту

При вызове скрипта параметры передаются по принципу "Call by Reference [Обращение по ссылке]". При использовании тега в качестве параметра операции присвоения значений в скрипте немедленно изменяют значение тега.

Однако вам нет необходимости в качестве параметров определять локальные теги в скрипте. Вместо этого вы можете непосредственно передавать значения.

Пример: системная функция "SetValue(Y, X)" присваивает значение "5" тегу "IndexTag":  
SetValue IndexTag, 5

#### Выходные параметры скрипта

Возвращаемые скриптом значения могут представлять собой результат выполнения вычислений (например определение среднего значения двух чисел). Кроме того, возвращаемые значения могут информировать о том, правильно ли была выполнена инструкция.

Поэтому системные функции, которые выполняют операции над файлами, например, "Delete [Удалить]" также могут возвращать значения, то есть иметь выходные параметры.

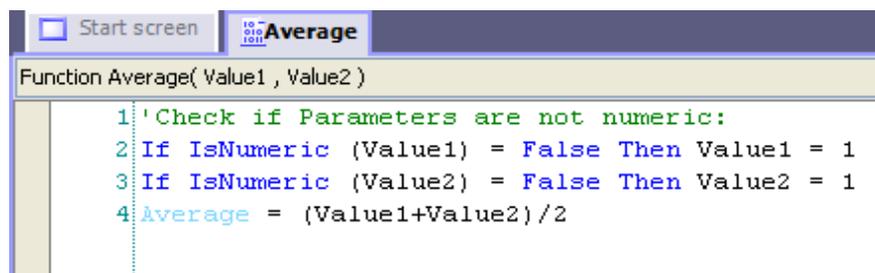
---

**Замечание**

Значение, возвращаемое системной функцией, может быть присвоено только внешнему или внутреннему тегу.

---

Для того чтобы скрипт имел возвращаемое значение (выходной параметр), необходимо выбрать в качестве типа скрипта тип "Function [Функция]". В таком скрипте значение выходного параметра, то есть возвращаемое значение присваивается имени скрипта:



```
Function Average( Value1 , Value2 )  
1 'Check if Parameters are not numeric:  
2 If IsNumeric (Value1) = False Then Value1 = 1  
3 If IsNumeric (Value2) = False Then Value2 = 1  
4 Average = (Value1+Value2)/2
```

Для получения среднего значения двух чисел вызовите функцию "Average" и передайте значения чисел тегу, например,:

Average Value = Average 4. 6

Среднее значение затем может быть выведено в поле вывода.

## 12.6.4 Изменение свойств объекта среды исполнения с помощью VBS

### Введение

С помощью VBS вы можете иметь доступ к свойствам экранных объектов и тегам среды исполнения. При изменении свойств объектов с помощью VBS эти изменения не оказывают влияния на проектные данные.

### Изменение свойств объекта

При изменении свойства экранного объекта в режиме исполнения проекта с помощью VBS это изменение действует только в течение того времени, пока экран остается активным. Как только вы переходите к другому экрану или перезагружаете этот же экран, будут вновь отображены свойства, указанные в проектных данных.

### Изменение языка

При изменении языка в режиме исполнения проекта тексты на различных языках загружаются из проектных данных. При смене языка с помощью VBS эти тексты заменяются на тексты на вновь выбранном языке.

## 12.6.5 Использование в скриптах системных функций, зависящих от устройства HMI

### Основные сведения

При использовании в скрипте системной функции, не поддерживаемой выбранным операторским устройством, будет выведено предупреждающее сообщение. Кроме того, соответствующая функция в скрипте будет подчеркнута волнистой синей линией.

