

SIEMENS

SIMATIC HMI

WinCC V7.0 ANSI-C для создания функций и макросов

Printout of the Online Help

Создание функций и макросов с помощью ANSI-C	1
Создание функций и макросов	2
Характеристики функций проекта	3
Характеристики стандартных функций	4
Характеристики внутренних функций	5
Характеристики локальных макросов	6
Характеристики глобальных макросов	7
Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта	8
Использование глобальных C-тегов	9
Использование DLL в функциях и макросах	10
Редактор Global Script	11
Создание и редактирование функций	12
Создание и редактирование макросов	13
Поведение макросов в среде исполнения	14

Правила техники безопасности

В этом руководстве содержатся примечания, которые необходимо соблюдать для обеспечения личной безопасности и предотвращения материального ущерба. В этом руководстве примечания, относящиеся к личной безопасности, обозначены символом предупреждения о безопасности; примечания, касающиеся только материального ущерба, не обозначаются символом предупреждения о безопасности. Эти перечисленные ниже примечания различаются по степени опасности.

⚠ ОПАСНО
Означает, что невыполнение надлежащих мер безопасности повлечет за собой смерть или серьезную травму.
⚠ ПРЕДУПРЕЖДЕНИЕ
Означает, что невыполнение надлежащих мер безопасности может повлечь за собой смерть или серьезную травму.
⚠ ВНИМАНИЕ!
При наличии символа предупреждения о безопасности это примечание означает, что невыполнение надлежащих мер безопасности может повлечь за собой несерьезные травмы.
ВНИМАНИЕ!
При отсутствии символа предупреждения о безопасности это примечание означает, что невыполнение надлежащих мер безопасности может повлечь за собой материальный ущерб.
ПРИМЕЧАНИЕ
Означает, что игнорирование соответствующей информации может привести к нежелательным результатам или последствиям.

При возникновении нескольких степеней опасности используется предупреждение, относящееся к наивысшей степени опасности. Предупреждение, относящееся к возможным травмам, с символом предупреждения о безопасности может также содержать предупреждение о возможном материальном ущербе.

Квалифицированный персонал

Установка и использование устройства/системы должны осуществляться исключительно в соответствии с настоящей документацией. Ввод устройства/системы в эксплуатацию и их эксплуатацию должен выполнять только **квалифицированный персонал**. В контексте правил техники безопасности настоящей документации под квалифицированным персоналом подразумеваются лица, которым разрешено осуществлять ввод в эксплуатацию, заземление и маркировку устройств, систем и электрических цепей в соответствии с установленными правилами и стандартами по технике безопасности.

Использование по назначению

Обратите внимание на следующие моменты.

⚠ ПРЕДУПРЕЖДЕНИЕ
Это устройство можно использовать только в целях, указанных в каталоге или техническом описании, и только совместно с устройствами либо компонентами сторонних производителей, одобренных или рекомендуемых компанией Siemens. Для обеспечения правильной, надежной эксплуатации продукта требуется надлежащая транспортировка, хранение, расположение и сборка, а также аккуратная эксплуатация и техническое обслуживание.

Торговые знаки

Все названия, отмеченные знаком ®, являются зарегистрированными торговыми знаками компании Siemens AG. Прочие торговые знаки, упомянутые в настоящем документе, могут быть торговыми знаками соответствующих владельцев. Использование таких торговых знаков третьими лицами в собственных целях может быть нарушением прав этих владельцев.

Отказ от ответственности

Мы проверили содержимое настоящего документа, чтобы убедиться в том, что оно соответствует описанному в нем оборудованию и программному обеспечению. Поскольку невозможно предусмотреть все изменения, полное соответствие не гарантируется. Тем не менее компания осуществляет проверку информации, представленной в настоящем документе, а также вносит все необходимые исправления в последующие издания.

Содержание

1	Создание функций и макросов с помощью ANSI-C	5
2	Создание функций и макросов.....	7
3	Характеристики функций проекта	11
4	Характеристики стандартных функций.....	13
5	Характеристики внутренних функций	15
6	Характеристики локальных макросов	17
7	Характеристики глобальных макросов	19
8	Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта.....	21
9	Использование глобальных C-тегов.....	23
10	Использование DLL в функциях и макросах	25
11	Редактор Global Script.....	27
11.1	Редактор Global Script.....	27
11.2	Работа в окне редактирования	30
11.2.1	Работа в окне редактирования	30
11.2.2	Редактирование функций с помощью клавиатуры	31
11.2.3	Редактирование функций с помощью мыши	32
11.3	Работа с панелями инструментов	32
11.4	Настройка разных представлений.....	35
11.5	Настройка стиля шрифта	35
11.6	Использование команды Save As... (Сохранить как...)	36
11.7	Удаление макросов или функций проекта и стандартных функций.....	36
11.8	Создание нового заголовка.....	37
11.9	Компилирование всех функций.....	38
11.10	Поиск в файлах	39
11.11	Печать функций и макросов.....	39
11.11.1	Печать функций и макросов.....	39
11.11.2	Настройка параметров печати.....	40
11.11.3	Открытие предварительного просмотра.....	41
11.11.4	Печать документации проекта	41
12	Создание и редактирование функций	43
12.1	Создание и редактирование функций.....	43
12.2	Создание новой функции	45
12.3	Запись кода функции	47
12.4	Использование внутренних функций.....	48

12.5	Использование стандартных функций и функций проекта	49
12.6	Вставка дополнительной информации о функции	50
12.7	Защита функции от несанкционированного доступа	51
12.8	Компилирование и сохранение функции	52
12.9	Переименование функции	53
12.10	Использование функций из других источников	54
13	Создание и редактирование макросов	57
13.1	Создание и редактирование макросов.....	57
13.2	Правило кодирования WinCC.....	60
13.3	Создание нового макроса.....	61
13.4	Редактирование макросов.....	61
13.5	Вставка дополнительной информации о макросе.....	62
13.6	Защита макроса от несанкционированного доступа.....	64
13.7	Компилирование и сохранение макроса	65
13.8	Триггеры.....	66
13.8.1	Триггеры.....	66
13.8.2	Добавление нового триггера типа Timer (Таймер).....	69
13.8.3	Добавление нового триггера типа Tag (Тег)	71
13.8.4	Изменение триггеров	72
13.8.5	Удаление триггера	74
13.9	Назначение разрешений	75
13.10	Экспорт макроса	76
13.11	Импорт макроса.....	77
13.12	Переименование макроса	77
13.13	Использование макросов из других источников.....	79
14	Поведение макросов в среде исполнения	81
14.1	Поведение макросов в среде исполнения	81
14.2	GSC Runtime (Среда исполнения макросов)	82
14.2.1	GSC Runtime (Среда исполнения макросов)	82
14.2.2	Помещение GSC Runtime (Среда исполнения макросов) в кадр процесса.....	85
14.2.3	Атрибуты GSC Runtime (Среда исполнения макросов).....	86
14.2.4	Редактирование макросов.....	86
14.3	GSC Diagnose (Диагностика макросов)	87
14.3.1	GSC Diagnose (Диагностика макросов)	87
14.3.2	Помещение GSC Diagnose (Диагностика макросов) в кадр процесса	88
14.3.3	Атрибуты GSC Diagnose (Диагностика макросов).....	89
14.3.4	Панель инструментов GSC Diagnose (Диагностика макросов).....	89
	Индекс	91

Создание функций и макросов с помощью ANSI-C

1

Содержание

Фоновые задачи, такие как печать ежедневных отчетов, мониторинг тегов или выполнение ориентированных на конкретные кадры расчетов, в среде исполнения осуществляются с помощью макросов.

Эти макросы запускаются триггерами.

Функции можно вызывать из макросов. WinCC имеет множество функций, которые могут быть изменены пользователем. Кроме того, пользователь может разработать собственные функции.

Редактор Global Script используется для создания и редактирования функций и макросов.

В данной главе содержится описание следующих процессов:

- Использование редактора Global Script
- Создание и редактирование функций
- Создание и редактирование макросов
- Использование инструментов диагностики для анализа проблем среды исполнения

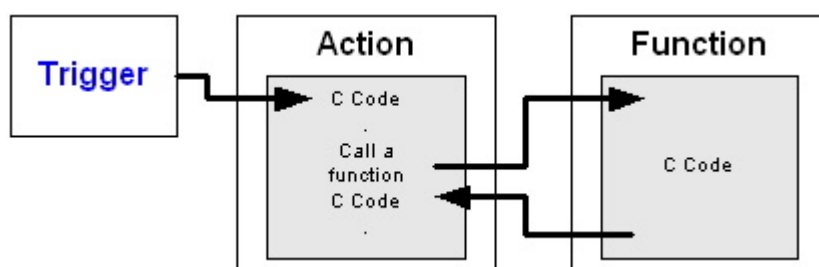
Создание функций и макросов

Введение

Система WinCC поддерживает использование функций и макросов для динамизации процессов в проекте WinCC. Эти функции и макросы записаны в ANSI-C.

Различие между функциями и макросами

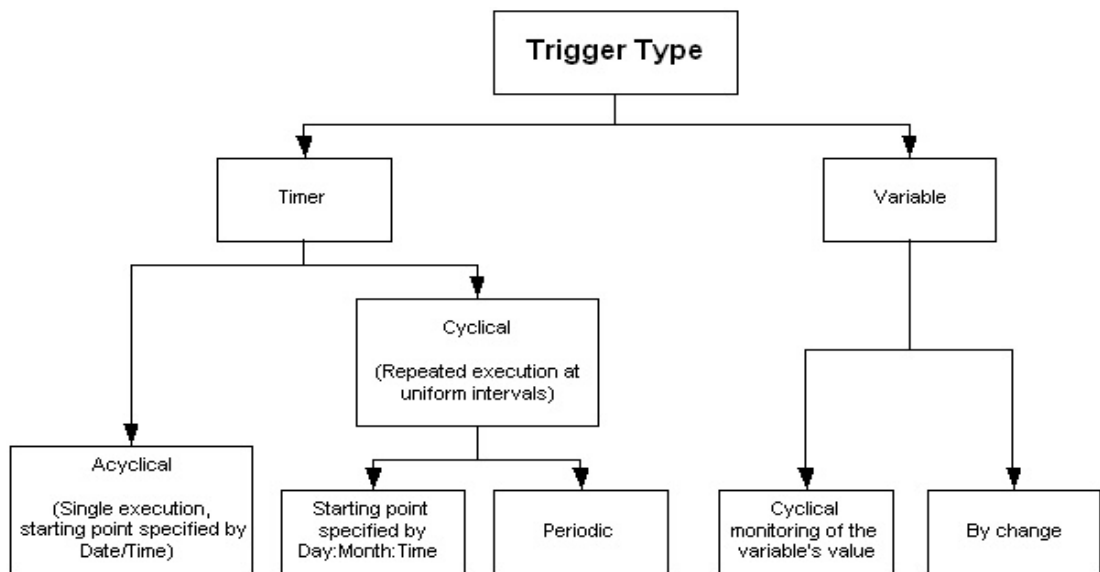
Макросы включаются триггером, то есть инициирующим событием. У функций нет триггера, они используются в качестве компонентов макросов, так же как в диалоговых окнах динамики, системе архивации тегов и системе регистрации аварийных сигналов.



Trigger	Триггер
Action	Макрос
C Code	Код-С
Call a function	Вызов функции
C Code	Код-С
Function	Функция
C Code	Код-С

Типы триггеров

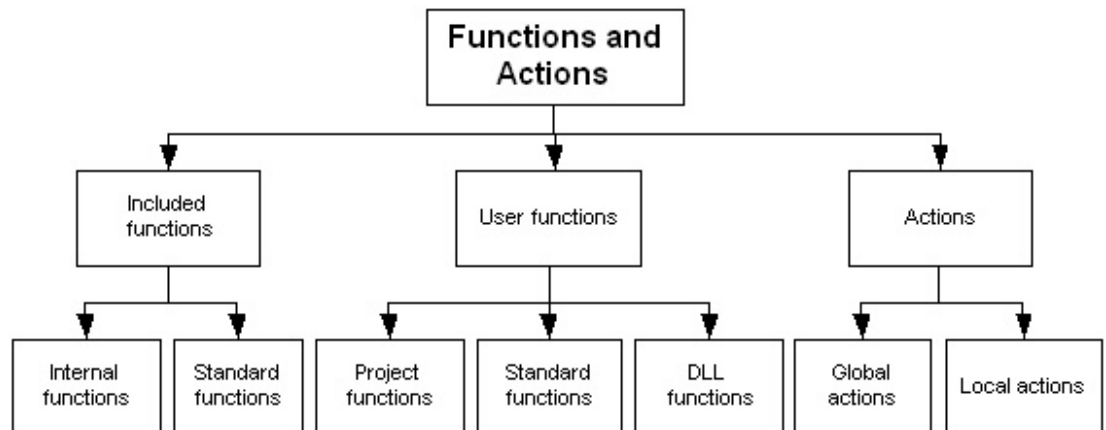
Доступны следующие типы триггеров:



Trigger Type	Тип триггера
Timer	Таймер
Variable	Переменная
Cyclical (Repeated execution at uniform intervals)	Циклический (повторяющееся выполнение с одинаковыми интервалами)
Acyclical (Single execution, starting point specified by Date/Time)	Нециклический (однократное выполнение, начальная точка указана в формате «дата/время»)
Starting point specified by Day:Month:Time	Начальная точка указана в формате «день:месяц:время»
Periodic	Периодический
Cyclical monitoring of the variable's value	Циклический мониторинг значения переменной
By change	При изменении

Обзор функций и макросов

На схеме представлен обзор ряда функций и макросов:



Functions and Actions	Функции и макросы
Included functions	Включенные функции
User functions	Функции пользователя
Actions	Макросы
Internal functions	Внутренние функции
Standard functions	Стандартные функции
Project functions	Функции проекта
Standard functions	Стандартные функции
DLL functions	Функции DLL
Global actions	Глобальные макросы
Local actions	Локальные макросы

Макросы используются для выполнения фоновых задач, не зависящих от кадра, таких как печать ежедневных отчетов, мониторинг тегов или выполнение расчетов.

Функции являются элементами кода, которые могут использоваться в нескольких местах, но определяются только в одном. WinCC включает в себя множество функций. Кроме того, можно записывать собственные функции и макросы.

Стандартные функции могут быть изменены пользователем. В случае переустановки или обновления WinCC измененные стандартные функции удаляются или заменяются неотредактированными стандартными функциями. Следует выполнить резервное копирование функций, измененных до обновления или переустановки.

Инструмент проектирования

WinCC поддерживает проектирование, создание и редактирование функций и макросов с помощью редактора Global Script. Редактор Global Script запускается из окна навигации проводника WinCC.

Дополнительные источники информации

- Поведение макросов в среде исполнения (стр. 97)
- Создание и редактирование макросов (стр. 71)
- Создание и редактирование функций (стр. 55)
- Редактор Global Script (стр. 37)
- Использование DLL в функциях и макросах (стр. 35)
- Использование глобальных C-тегов (стр. 31)
- Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта (стр. 27)
- Создание нового заголовка (стр. 48)
- Характеристики глобальных макросов (стр. 23)
- Характеристики локальных макросов (стр. 19)
- Характеристики внутренних функций (стр. 17)
- Характеристики стандартных функций (стр. 15)
- Характеристики функций проекта (стр. 13)

Характеристики функций проекта

Характеристики функций проекта

Функции проекта...

- могут быть созданы пользователем;
- могут быть отредактированы пользователем;
- могут быть защищены паролем от изменения и просмотра лицами, не имеющими доступа;
- не имеют триггера;
- объявляются только в рамках проекта;
- получают расширение имени файла "*.fct".

Функции проекта сохраняются в подкаталоге "library" проекта WinCC.



Использование функций проекта

Функции проекта могут быть использованы...

- в других функциях проекта;
- в макросах Global Script;
- в C-макросах в графическом дизайнере и в диалоговом окне динамики;
- в функции показа контекста системы регистрации аварийных сигналов;
- в системе архивации тегов во время запуска и разблокировки архивов, а также при выгрузке циклических архивов.

Дополнительные источники информации

Защита функции от несанкционированного доступа (стр. 64)

Создание и редактирование функций (стр. 55)

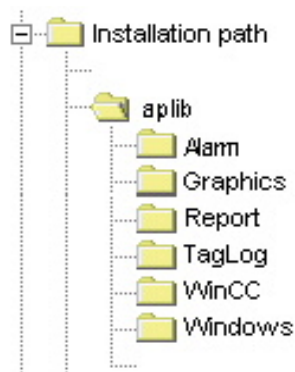
Характеристики стандартных функций

Характеристики стандартных функций

Стандартные функции...

- доступны для использования в WinCC;
- не могут быть созданы пользователем;
- могут быть отредактированы пользователем;
- могут быть защищены паролем от изменения и просмотра лицами, не имеющими доступа;
- не имеют триггера;
- объявляются в нескольких разных проектах;
- получают расширение имени файла «*.fct».

Стандартные функции сохраняются в подкаталогах «\aplib» в каталоге установки WinCC.



Использование стандартных функций

Стандартные функции могут быть использованы...

- в функциях проекта;
- в других стандартных функциях;
- в макросах Global Script;
- в C-макросах в графическом дизайнера и в диалоговом окне динамики;
- в функции показа контекста системы регистрации аварийных сигналов;

- в системе архивации тегов во время запуска и разблокировки архивов, а также при выгрузке циклических архивов.

Примечание

Прилагающиеся стандартные функции могут быть отредактированы пользователем. В случае переустановки или обновления WinCC измененные стандартные функции удаляются или заменяются неотредактированными стандартными функциями. Следует выполнить резервное копирование функций, измененных до обновления или переустановки.

Дополнительные источники информации

Использование стандартных функций и функций проекта (стр. 62)

Создание и редактирование функций (стр. 55)

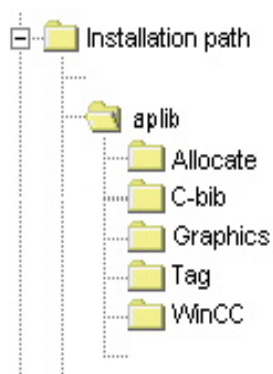
Характеристики внутренних функций

Характеристики внутренних функций

Внутренние функции ...

- доступны для использования в WinCC;
- **не могут быть** созданы пользователем;
- **не могут быть** редактированы;
- **не могут быть** переименованы;
- не имеют триггера;
- доступны в рамках всего проекта;
- получают расширение имени файла «*.fct».

Внутренние функции сохраняются в подкаталогах «\aplib» в каталоге установки WinCC.



Использование внутренних функций

Внутренние функции могут быть использованы...

- в функциях проекта;
- в стандартных функциях;
- в макросах;
- в C-макросах в графическом дизайнера и в диалоговом окне динамики.

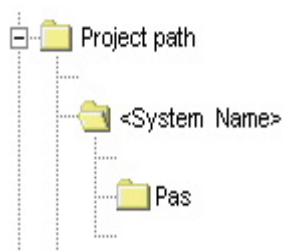
Характеристики локальных макросов

Характеристики локальных макросов

Локальные макросы...

- могут быть созданы пользователем;
- могут быть отредактированы пользователем;
- могут быть защищены паролем от изменения и просмотра лицами, не имеющими доступа;
- имеют по меньшей мере один триггер;
- используются только на назначенном компьютере;
- получают расширение имени файла «*.fct».

Локальные макросы сохраняются в подкаталоге «\<computer_name\Pas» в каталоге проекта.



Использование локальных макросов

Макросы используются для выполнения фоновых задач, не зависящих от кадра, таких как печать ежедневных отчетов, мониторинг тегов или выполнение расчетов. Макрос запускается заданным для него триггером. Для выполнения макроса необходимо, чтобы Global Script Runtime (Среда исполнения Global Script) была включена в список запуска.

В отличие от глобальных макросов, локальные макросы можно назначить для одного компьютера. Таким образом, можно, например, обеспечить печать отчета только на сервере.

Дополнительные источники информации

Защита макроса от несанкционированного доступа (стр. 79)

Триггеры (стр. 81)

Создание и редактирование макросов (стр. 71)

Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта (стр. 27)

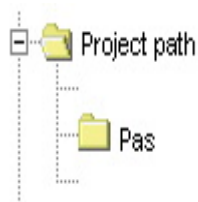
Характеристики глобальных макросов

Характеристики глобальных макросов

Глобальные макросы...

- могут быть созданы пользователем;
- могут быть отредактированы пользователем;
- могут быть защищены паролем от изменения и просмотра лицами, не имеющими доступа;
- имеют по меньшей мере один триггер для запуска;
- выполняются на всех компьютерах в проекте клиент/сервер;
- получают расширение имени файла «*.fct».

Глобальные макросы сохраняются в подкаталоге «\Pas» проекта WinCC.



Использование глобальных макросов

Макросы используются для выполнения фоновых задач, таких как печать ежедневных отчетов, мониторинг тегов или выполнение расчетов. Макрос запускается заданным для него триггером. Для выполнения макроса необходимо, чтобы Global Script Runtime (Среда исполнения Global Script) была включена в список запуска.

В отличие от локальных макросов, глобальные макросы выполняются на всех компьютерах в проекте клиент/сервер. В однопользовательском проекте различия между глобальными и локальными макросами не существует.

Дополнительные источники информации

Защита макроса от несанкционированного доступа (стр. 79)

Триггеры (стр. 81)

Создание и редактирование макросов (стр. 71)

Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта (стр. 27)

Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта

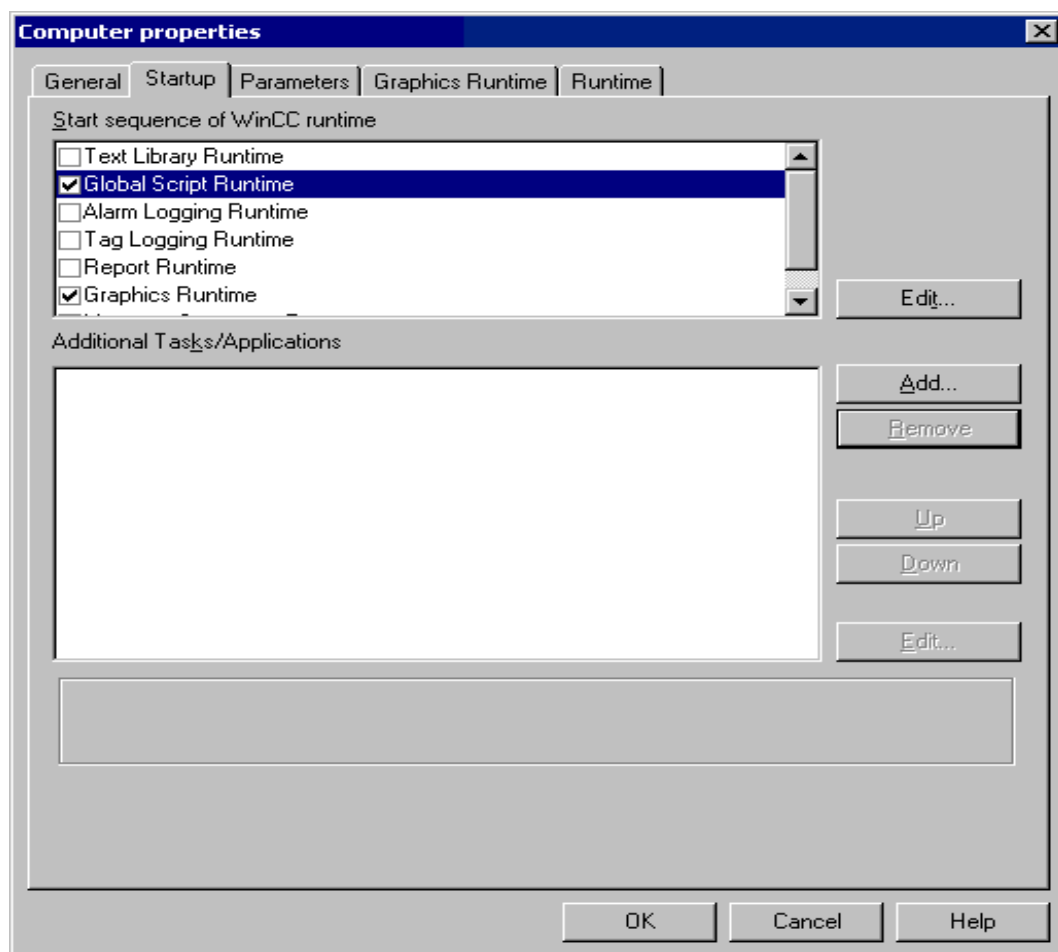
8

Введение

Для использования макросов Global Script в среде исполнения необходимо, чтобы Global Script Runtime (Среда исполнения Global Script) была добавлена в список запуска проекта. На выполнение функций это не повлияет.

Процедура

1. В контекстном меню компьютера в проводнике WinCC выберите Properties (Свойства). Откроется диалоговое окно Computer Properties (Свойства компьютера).
2. Перейдите на вкладку Startup (Запуск).
3. Выберите параметр Global Script Runtime (Среда исполнения Global Script).



4. Нажмите кнопку ОК, чтобы закрыть диалоговое окно.

Использование глобальных C-тегов

Определение глобальных C-тегов

Глобальный C-тег определяется путем добавления строки определения перед именем функции:

```
int a; //Тег a определяется как целое число
void dummy() //Имя функции
{
. //Код функции
}
```

Диапазон применимости

Тег, заданный таким образом, объявляется для каждой функции и макроса в среде исполнения. Он создается сразу при запуске среды исполнения, даже если функция не была вызвана.

Использование глобальных C-тегов

Глобальные C-теги используются в функциях или макросах путем их объявления как внутренних:

```
void dummy() //Имя функции
{
extern int a; //Внешнее объявление тега a
. //Код функции
}
```

Таким образом, компилятору отдается команда не создавать тег, так как он создан в другом местоположении в среде исполнения.

Если значение тега a изменяется, это изменение может быть прочитано каждой функцией и макросом.

Каждый C-тег может быть определен только в одном местоположении. Для ясности и во избежание дублирования определений рекомендуется определять глобальные C-теги только в одном местоположении.

Примечание

Для функции и определяемого с помощью нее глобального C-тега доступно максимум 64 Кб.

Использование DLL в функциях и макросах

Настройка DLL

WinCC позволяет использовать собственные DLL (динамически подключаемые библиотеки).

Функции в существующих DLL можно определять для функций и макросов путем внесения необходимых дополнений в соответствующую функцию или макрос.

Добавьте следующий код перед функцией или макросом:

```
#pragma code("<Name>.dll")
<Type of returned value> <Function_name 1>(...);
<Type of returned value> <Function_name2>(...);
.
.
.
<Type of returned value> <Function_name n>(...);
#pragma code()
```

Функции с <Function_name 1> по <Function_name n> из <Name.dll> объявлены и теперь могут быть вызваны соответствующей функцией или макросом.

Пример.

```
#pragma code("kernel32.dll")
VOID GetLocalTime(LPSYSTEMTIME lpSystemTime);
#pragma code()

SYSTEMTIME st;

GetLocalTime(&st);
```

Существует альтернативная процедура – необходимые дополнения можно вносить в заголовочный файл Apdefap.h.

При применении собственных DLL в WinCC необходимо использовать окончательную версию. WinCC поставляется в окончательной версии и, следовательно, использует окончательную версию системных DLL. При создании пользовательской DLL в отладочной версии могут быть загружены DLL как окончательной, так и отладочной версии, что приведет к повышению нагрузки на память.

Структуры DLL должны быть настроены с использованием однобайтового выравнивания.

Примечание

DLL необходимо сохранить в каталоге «\bin» или в каталоге, путь к которому определен в системном теге PATH. Этот тег определяется в свойствах операционной системы.

Редактор Global Script

11.1 Редактор Global Script

Введение

WinCC позволяет создавать и редактировать функции и макросы с помощью редактора Global Script. Редактор Global Script запускается из окна проекта в проводнике WinCC.

Структура редактора Global Script

Редактор Global Script спроектирован в соответствии со стандартами Windows. В нем представлены панели инструментов, строки меню и строки состояния. В редакторе также имеется несколько окон со раскрывающимися меню.

Рисунок 11-1 Редактор Global Script

Окно навигации

Окно навигации используется для выбора функций и макросов, которые требуется отредактировать или вставить в местоположении курсора в окне редактирования.

Функции и макросы имеют иерархическую систему групп.

Для отображения функций используются их имена. Для макросов же используются имена файлов.

Окно редактирования

Функции и макросы можно записывать и редактировать в окне редактирования. Оно отображается только при открытии функции или макроса с целью редактирования. Каждая функция или макрос открывается в отдельном окне редактирования. Одновременно можно открыть несколько окон.

Окно вывода

Результаты функций Find in Files (Найти в файлах) или Compile All Functions (Компилировать все функции) отображаются в окне вывода. Оно открывается по умолчанию, но может быть скрыто.

Find in Files (Найти в файлах):

Результаты поиска отображаются в окне вывода построчно. Каждая строка имеет номер, и в ней содержится путь к файлу и его имя, а также номер и текст строки, в которой был найден искомый текст.

Связанный файл можно открыть путем двойного нажатия строки, отображаемой в окне вывода. Курсор помещается на строку, в которой был найден искомый элемент.

Compile All Functions (Компилировать все функции):

При необходимости компилятор выводит предупреждения и сообщения об ошибках для каждой скомпилированной функции. В следующей строке отображается путь к файлу скомпилированной функции и его имя, а также сводное сообщение из компилятора.

Строка меню

Кнопки строки меню различаются в зависимости от ситуации. Строка состояния всегда отображается.

Панели инструментов

В Global Script имеются две панели инструментов. При необходимости их можно отобразить и с помощью мыши переместить в любое положение на экране.

Строка состояния

Строка состояния расположена в нижней части окна Global Script. Ее можно отобразить или скрыть. В ней содержится информация о положении курсора в окне редактирования и настройках клавиатуры. Кроме того, в строке состояния отображается краткое описание выбранной в данный момент функции Global Script или подсказка.

Дополнительные источники информации

Печать функций и макросов (стр. 50)

Поиск в файлах (стр. 50)

Компилирование всех функций (стр. 49)

Создание нового заголовка (стр. 48)

Удаление макросов или функций проекта и стандартных функций (стр. 48)

Использование команды Save As... (Сохранить как...) (стр. 47)

Настройка стиля шрифта (стр. 46)

Настройка разных представлений (стр. 46)

Работа с панелями инструментов (стр. 43)

Работа в окне редактирования (стр. 40)

11.2 Работа в окне редактирования

11.2.1 Работа в окне редактирования

Введение

В окне редактирования содержится ряд функций, работать с которыми можно как с помощью клавиатуры, так и с помощью мыши.

В окне редактирования можно редактировать функции и макросы.

```

Test01.mcp : autoload.pas
#include "apdefap.h"

int gscAction( void )
{
#pragma option (mbcs)

char* pUserName = NULL;

// Try to get currently logged in user.
pUserName = GetTagChar ("@local::@CurrentUser");

// Load Default SSM of current user.
// if pUserName == NULL or contains an empty-string, the global Default.SSM is restored.
if (FALSE == SSMLoadCurrentFields ("Default", pUserName))
    if (FALSE == SSMLoadCurrentFields ("Default", ""))
        return FALSE;

return TRUE;
}

Compiling...
0 Error(s), 0 Warning(s)
    
```

Содержимое окна

Окно можно разделить. В верхней части окна редактирования отображается код функции или макроса. В нижней части содержатся сообщения, выводимые компилятором во время компилирования функции или макроса.

Свойства

При первом открытии окна его нижняя часть сворачивается. При запуске процесса компиляции нижняя часть окна разворачивается, и в ней отображаются сообщения компилятора. Положение разделителя в окне можно настроить с помощью мыши.

Чтобы перейти на соответствующую строку в коде, дважды щелкните сообщение об ошибке.

Цветовой код

Код-С имеет следующие цветовые коды:

Цвет	Описание	Пример
Синий	Ключевые слова	define, double, if
Зеленый	Комментарии	// комментарий
Красный	Строки	Rectangle3
Черный	Другие С-коды	level=100*newvalue/255;

Примечание

Функция или макрос не может иметь больше 32767 символов, включая пробелы.

Дополнительные источники информации

Редактирование функций с помощью мыши (стр. 42)

Редактирование функций с помощью клавиатуры (стр. 41)

11.2.2 Редактирование функций с помощью клавиатуры

С помощью клавиатуры можно выполнять следующие операции редактирования функций:

Редактирование функции	Управление клавиатурой
Переключение между режимами записи: вставка/перезапись	<INSERT>
Добавление новой строки	<ВВОД>
Удаление одного символа справа	<DELETE>
Удаление одного символа слева	<BACKSPACE>
Удаление выделенного текста	<DELETE> или <BACKSPACE>
Переход к началу строки	<POS1>
Переход к концу строки	<END>
Переход к началу текста	<CTRL+POS1>
Переход к концу текста	<CTRL+END>
Перемещение курсора	<Клавиши курсора>
Перемещение курсора на один экран вверх к началу текста	<PAGE UP>
Перемещение курсора на один экран вниз к концу текста	<PAGE DOWN>
Перемещение курсора к первой строке в окне	<CTRL+PAGE UP>

Редактирование функции	Управление клавиатурой
Перемещение курсора к последней строке в окне	<CTRL+PAGE DOWN>
Переход к следующему объекту окна по клавише табуляции	<TAB>
Вырезание и вставка выделенного текста в буфер обмена	<CTRL+X>
Копирование выделенного текста в буфер обмена	<CTRL+C>
Вставка текста из буфера обмена	<CTRL+V>

11.2.3 Редактирование функций с помощью мыши

С помощью мыши можно выполнять следующие операции редактирования функций:

Редактирование функции	Команда, выполняемая мышью (левая кнопка мыши)
Выделение текста	Обвести текст курсором, нажав и удерживая кнопку мыши
Выбор слова	Дважды щелкнуть слово
Выбор строки	Трижды щелкнуть строку
Расширенный выбор	<SHIFT> + нажать кнопку мыши
Перемещение курсора	Нажать кнопку мыши
Перемещение выделенного текста	Перетащить выделенный текст
Дублирование выделенного текста	<CTRL>+перетащить выделенный текст

Другие функции редактирования:

- При двойном нажатии сообщения об ошибке компилятора редактор переходит к соответствующей строке в коде.
- При нажатии правой кнопки мыши вызывается контекстное меню.

При выполнении следующих действий выделенный текст заменяется результатом действия:

- Ввод символа(ов) с помощью клавиатуры
- Вставка текста из буфера обмена
- Добавление вызова функции путем назначения параметров

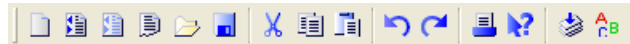
11.3 Работа с панелями инструментов

Предназначение

Панели инструментов располагаются по умолчанию под строкой меню в верхней части окна Global Script. Кнопки панелей инструментов предназначены для быстрого и легкого вызова ряда функций Global Script.

Существует две панели инструментов:

Стандартная панель


















Панель редактирования



Содержимое панелей инструментов

На стандартной панели представлены кнопки для вызова следующих функций:







Кнопка	Функция	Сочетание клавиш
	Создание нового макроса	<ALT+A> или <CTRL+N>
	Создание новой стандартной функции	<ALT+S> или <CTRL+N>
	Создание новой функции проекта	<ALT+P> или <CTRL+N>
	Создание нового заголовочного файла	<CTRL+H>
	Открытие существующей функции или макроса	<CTRL+O>
	Сохранение содержимого активного окна редактирования. Функция доступна только при открытом окне редактирования	<CTRL+S>
	Вырезание выбранного текста и его копирование в буфер обмена. Функция доступна, только если текст был выделен	<CTRL+X>
	Копирование выбранного текста в буфер обмена. Функция доступна, только если текст был выделен	<CTRL+C>
	Вставка содержимого буфера обмена в местоположении курсора. Функция доступна, только если в буфере обмена содержатся данные	<CTRL+V>
	Отмена минимум 30 последних действий редактора. Функция доступна, только если в редакторе было выполнено действие	<CTRL+Z>
	Повторение последнего отмененного действия редактора. Функция доступна, только если в редакторе было отменено действие	<CTRL+A>
	Печать содержимого активного окна редактирования как документации проекта. Функция доступна только при открытом окне редактирования	<CTRL+P>
	Отображение прямой справки (Что это такое?)	<SHIFT+F1>

Кнопка	Функция	Сочетание клавиш
	Создание нового заголовка	<CTRL+G>
	Настройка шрифта	<CTRL+F>

Примечание

Сочетание клавиш <CTRL+N> доступно, если открыто хотя бы одно окно редактирования. Если в активном окне редактирования содержится функция, то при нажатии <CTRL+N> создается новая функция проекта. Если в активном окне редактирования содержится макрос, это сочетание клавиш позволяет создать новый глобальный макрос.

На панели редактирования представлены кнопки для вызова следующих функций:

Кнопка	Функция	Сочетание клавиш
	Добавление информации о функциях, а в случае макроса – настройка триггера. Функция доступна только при открытом окне редактирования	<CTRL+I>
	Компиляция кода в активном окне редактирования. Функция доступна только при открытом окне редактирования	<SHIFT+F8>
	Открытие диалогового окна тегов. Функция доступна только при открытом окне редактирования	<CTRL+R>
	Открытие диалогового окна выбора кадра. Имя выбранного кадра вставляется в местоположении курсора в окне редактирования. Функция доступна только при открытом окне редактирования	<CTRL+W>
	Импорт макроса. Данная функция доступна только при наличии макроса в активном окне	<CTRL+M>
	Экспорт макроса из активного окна редактирования. Данная функция доступна только при наличии макроса в активном окне	<CTRL+T>

Свойства

Обе панели инструментов можно отображать и скрывать.

Их можно закреплять под строкой меню.

Если они не прикреплены, их можно перетащить в любое положение на экране с помощью мыши.

Дополнительные источники информации

Настройка разных представлений (стр. 46)

11.4 Настройка разных представлений

Введение

В данном контексте представлениями называются различные сочетания элементов, отображаемых в редакторе Global Script, таких как окно вывода, строка состояния и панели инструментов. Эти элементы можно отображать или скрывать по отдельности.

Все элементы отображаются по умолчанию.

Процедура

1. В строке меню Global Script откройте меню View (Вид).
2. Отобразите или скройте необходимые элементы, например панели инструментов. Если выбрано "show" (показывать), перед именем отображается галочка.

Примечание

При повторном запуске Global Script в редакторе восстанавливаются настройки по умолчанию, и все элементы снова отображаются.


11.5 Настройка стиля шрифта

Введение

Для стиля шрифта доступны следующие настройки: Font (Шрифт), Style (Стиль) и Size (Размер).

Выбранный стиль применяется во всех окнах редактирования.

Процедура

1. Нажмите кнопку  на стандартной панели инструментов, чтобы открыть диалоговое окно для настройки стиля шрифта.
2. Выполните необходимые настройки.
3. Подтвердите настройки, нажав ОК.

Альтернативная процедура

Диалоговое окно с настройками стиля шрифта можно также открыть следующим способом:

В строке меню Global Script откройте меню Options (Параметры) и выберите Font (Шрифт) либо воспользуйтесь соответствующим сочетанием клавиш.

Примечание

Настройки сохраняются автоматически и не сбрасываются при повторном запуске WinCC.

11.6 Использование команды Save As... (Сохранить как...)

Введение

При создании функции или макроса в Global Script производится сохранение соответствующего файла в указанную папку с использованием имени файла по умолчанию, например new_function_1.fct для функций и gscs1.pas для макросов. Так как эти имена файла по умолчанию не особо пригодны, используйте Save As ... (Сохранить как...) для сохранения функции или макроса под другим (более понятным) именем файла. Файл с именем по умолчанию остается.

При использовании Save As ... (Сохранить как...) изменяется только имя файла, имя функции или макроса остается неизменным.

В Global Script функции и макросы по умолчанию сохраняются в каталоге проекта. В противном случае отображается сообщение, однако файл все равно сохраняется.

Требования

Команда Save As... (Сохранить как...) доступна, если открыто хотя бы одно окно редактирования. Она позволяет сохранить содержимое активного окна редактирования.

Процедура

1. В строке меню Global Script откройте меню File (Файл).
2. Выберите Save As... (Сохранить как...).
3. Введите новое имя файла.
4. Закройте диалоговое окно, нажав кнопку Save (Сохранить).

Дополнительные источники информации

Удаление макросов или функций проекта и стандартных функций (стр. 48)

11.7 Удаление макросов или функций проекта и стандартных функций

Введение

Макросы и функции проекта, а также стандартные функции можно удалять во время настройки или в среде исполнения. В Global Script удаляется запись в окне навигации, а также ассоциированный файл.

Если удаленная функция вызывается макросом, макрос останавливается при вызове функции.

Если в это время открыто окно диагностики Global Script, отобразится сообщение. Остановка макроса регистрируется в файле диагностики WinCC_Sys_xx.log (xx = порядковый номер). Файл диагностики находится в подкаталоге Diagnostics (Диагностика) в каталоге установки WinCC.

Процедура

1. В окне навигации Global Script вызовите контекстное меню для функции или макроса, который требуется удалить.
2. Выберите пункт Delete (Удалить).
3. Подтвердите команду нажатием Yes (Да).

Альтернативная операция

Вместо использования контекстного меню выбранную функцию или макрос можно удалить при помощи клавиши <DELETE>.

Примечание

Если функция удаляется, запись в соответствующем заголовочном файле удаляется вместе с ней.

11.8 Создание нового заголовка

Введение

Заголовок повторно создается в следующих случаях:


- После копирования функций проекта из другого проекта в подкаталог library (библиотека) каталога проекта.
- После копирования стандартных функций с другого ПК в каталог (или подкаталог) arlib.

При повторном создании заголовка скопированные функции добавляются в соответствующие заголовочные файлы и затем становятся доступными для использования в проекте.

Требования

Функция доступна, только если окна редактирования закрыты.

Процедура

1. Нажмите кнопку  на стандартной панели инструментов.

Альтернативная операция

Процесс создания можно также запустить следующим способом:

Откройте меню Options (Параметры) и выберите Regenerate Header (Повторно создать заголовок) или воспользуйтесь соответствующим сочетанием клавиш.

Примечание

После завершения повторного создания содержимое окна навигации обновляется.

Если WinCC находится в среде исполнения, повторное создание заголовков не повлияет на систему среды исполнения.

11.9 Компилирование всех функций

Введение

После внесения вручную всех изменений в заголовочные файлы необходимо повторно скомпилировать все функции. Все стандартные функции, внутренние функции и функции проекта компилируются автоматически.

Результаты отдельных операций компиляции отображаются в окне вывода следующим образом:

При необходимости компилятор выводит предупреждения и сообщения об ошибках для каждой компилируемой функции. В следующей строке отображается путь к файлу компилированной функции и его имя, а также сводное сообщение из компилятора.

Если функции вызываются другими функциями, могут возникнуть ошибки, в результате чего могут отобразиться сообщения об ошибках. Это происходит по причине того, что вызываемые функции еще не скомпилированы. Эти функции должны быть скомпилированы отдельно.

Требования

Функция доступна, только если окна редактирования закрыты.

Процедура

1. Откройте меню Options (Параметры).
2. Выберите Compile All Functions (Компилировать все функции).

Альтернативная операция

Все функции можно скомпилировать с помощью сочетания клавиш <ALT+U>.

Примечание

Функция Compile All Functions (Компилировать все функции) недоступна в многопользовательском проекте. Причиной этому служит то, что распределение функций в таком проекте больше невозможно.

11.10 Поиск в файлах

Введение

Во всех файлах группы, выбранной в окне навигации, производится поиск указанного элемента.

Результат поиска отображается в окне вывода следующим образом:

В окне вывода для каждого найденного искомого элемента отображается строка. Строка включает в себя номер строки в коде, в котором был найден искомый элемент, путь к файлу и его имя, а также саму строку кода.

Стандартные функции и функции проекта, так же как и макросы, можно открывать двойным нажатием результата поиска. Курсор помещается в начало строки, в которой был найден искомый элемент. При работе с внутренними функциями в окне навигации отображается и выбирается функция, содержащая искомый элемент.

Процедура

1. В окне навигации Global Script откройте контекстное меню для группы, в которой требуется выполнить поиск.
2. Выберите Find in Files (Найти в файлах):
3. В диалоговом окне введите искомый элемент.
4. Нажмите Find (Найти), чтобы начать поиск. Результат поиска отображается в окне вывода.

11.11 Печать функций и макросов

11.11.1 Печать функций и макросов

Введение

Макросы и функции проекта, а также стандартные функции можно печатать с помощью указанных системных макетов.

Однако это возможно, только если функция или макрос, который требуется распечатать, отображаются в окне редактирования. Печатается содержимое активного окна редактирования.

Распечатку можно проверить на экране в предварительном просмотре.

Процесс печати можно настроить с помощью нескольких параметров печати.

Используются следующие системные макеты:

- @gsc_pfc.rpl для функций проекта
- @gsc_pfc.rpl для стандартных функций
- @gsc_act.rpl для макросов

Дополнительные источники информации

Печать документации проекта (стр. 52)

Открытие предварительного просмотра (стр. 52)

Настройка параметров печати (стр. 51)

11.11.2 Настройка параметров печати

Введение

Печать можно настроить следующим способом:

- Указав макет, отличающийся от стандартного
- Выбрав диапазон страниц
- Выбрав принтер
- Выполнив печать в файл

Требования

Должно быть открыто хотя бы одно окно редактирования.

Процедура

1. В строке меню Global Script откройте меню File (Файл).
2. Выберите Project Documentation Setup... (Настройка документации проекта...)
3. В открывшемся диалоговом окне выполните необходимые настройки.
4. Примените настройки, нажав кнопку ОК.

Примечание

Настройки сохраняются автоматически и не сбрасываются при повторном запуске WinCC.

11.11.3 Открытие предварительного просмотра

Введение

Перед началом печати функции или макроса, иногда лучше предварительно просмотреть распечатку на экране (в предварительном просмотре).

Содержимое активного окна редактирования отображается в предварительном просмотре.

Процедура

1. В строке меню Global Script откройте меню File (Файл).
2. Выберите Project Documentation Setup... (Настройка документации проекта...).

11.11.4 Печать документации проекта

Введение

Содержимое активного окна редактирования можно вывести на принтер или в файл. При этом применяются настроенные параметры печати.

Процедура

1. В строке меню Global Script откройте меню File (Файл).
2. Выберите Print Project Documentation (Печать документации проекта)
- 3.

4.

Создание и редактирование функций

12.1 Создание и редактирование функций

Введение

В системе существует различие между функциями проекта, стандартными функциями и внутренними функциями. В WinCC представлен широкий выбор стандартных и внутренних функций. Кроме того, здесь можно создавать собственные проекты, а также создавать и изменять стандартные функции. Необходимо помнить, что стандартные функции, прилагаемые к WinCC, перезаписываются при переустановке WinCC, поэтому любые изменения теряются.

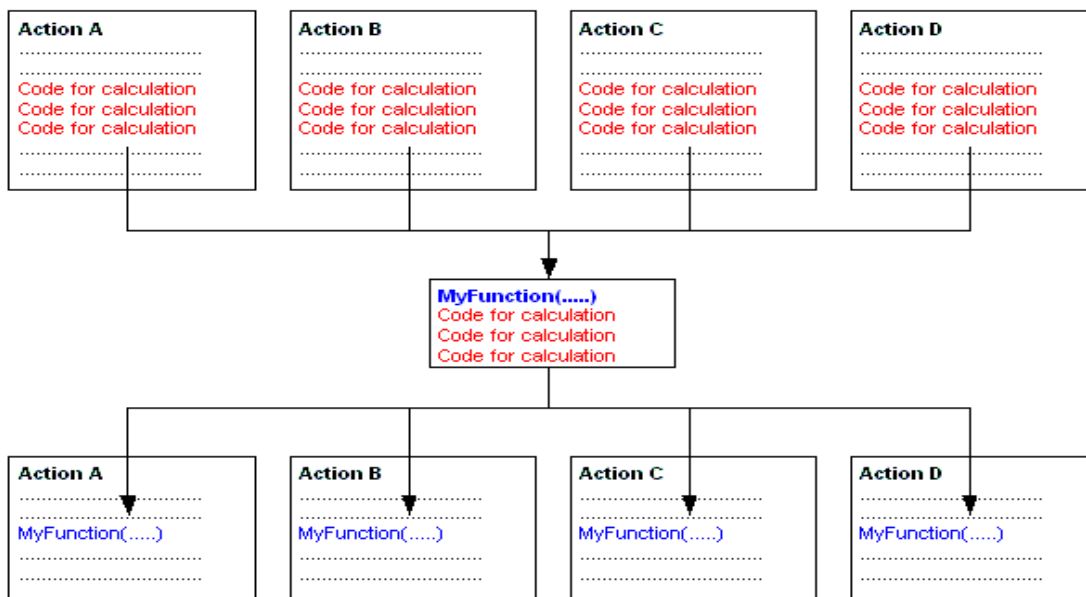
Внутренние функции недоступны для создания или редактирования.

Использование функций

Если требуется выполнить одинаковый расчет (с разными начальными значениями) в нескольких макросах, для этой цели будет выгоднее запрограммировать функцию. Впоследствии можно просто вызывать эту функцию с помощью текущих параметров в макросах.

Преимущества данного метода:

- Код вводится только один раз.
- Изменения вносятся только в одной точке, а именно в процедуре, а не в каждом макросе.
- Код макроса получается короче и проще.



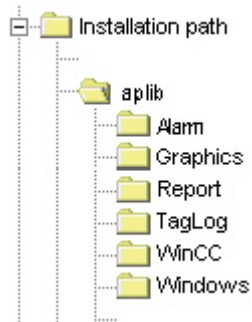
Поиск функций

Для доступа к существующим функциям или создания новых можно воспользоваться меню File (Файл) в окне навигации Global Script или нажать соответствующую кнопку на панели инструментов.

Функции хранятся в файловой системе следующим образом:



Функции проекта



Стандартные функции


Редактирование и компилирование функций


Редактирование и компилирование функции выполняется в отдельном окне редактирования. В окне редактирования отображаются сообщения компилятора после выполнения компиляции. Это могут быть предупреждения или сообщения об ошибках. В каждом из случаев выводится сводка о количестве предупреждений и сообщений об ошибках.

Изменения при переименовании функций

В окне навигации функции всегда отображаются со своим именем, а не с именем файла Windows. Если изменить имя функции и затем скомпилировать ее, имя, отображаемое в окне навигации, больше не будет совпадать с именем функции. На это в окне навигации указывает наличие префикса «*» (звездочка) перед именем. Сразу после сохранения функции текущее имя функции отображается в окне навигации.

Примечание относительно сохранения функций

При сохранении функции, которая не была скомпилирована должным образом, в окне навигации отображается значок .

При сохранении функции, которая была скомпилирована с ошибками, в окне навигации отображается значок .

Дополнительные источники информации

- Работа в окне редактирования (стр. 40)
- Использование функций из других источников (стр. 67)
- Переименование функции (стр. 66)
- Компилирование и сохранение функции (стр. 65)
- Защита функции от несанкционированного доступа (стр. 64)
- Вставка дополнительной информации о функции (стр. 63)
- Использование стандартных функций и функций проекта (стр. 62)
- Использование внутренних функций (стр. 61)
- Запись кода функции (стр. 60)
- Создание новой функции (стр. 58)
- Характеристики стандартных функций (стр. 15)
- Характеристики функций проекта (стр. 13)

12.2 Создание новой функции


Введение

Процедура является одинаковой как для функций проекта, так и для стандартных функций. В окне навигации укажите тип (функция проекта или стандартная функция). Для стандартных функций укажите группу, например graphics (графические). От этого также зависит, где будет храниться файл.

В Global Script предлагается использовать для новой функции имя по умолчанию, например new_function_3. Это имя также является именем файла функции. Для

обеспечения уникального имени функции в предлагаемом имени содержится порядковый номер.

Обычно имя по умолчанию стоит заменить более информативным именем функции. При первом сохранении переименованной функции можно также изменить имя файла.

В Global Script к функции добавляется следующая информация: дата создания, дата изменения и версия. Данную информацию можно просмотреть в диалоговом окне Properties (Свойства). В этом окне можно также назначить пароль для защиты функции от несанкционированного изменения и просмотра. Чтобы открыть диалоговое окно, нажмите кнопку .

Примечание

Обратите внимание, что в имени функции можно использовать только определенные символы: символы (за исключением специальных национальных знаков), цифры и подчеркивание.

Процедура

1. В окне навигации откройте контекстное меню для необходимой группы.
2. Выберите New (Создать).
При создании новой функции в первой строке кода в соответствующем окне редактирования содержатся данные о типе возвращаемого значения и имени новой функции по умолчанию. При необходимости в следующих скобках можно ввести формальный параметр.
Код функции вводится между фигурными скобками.

Альтернативная операция

Новую функцию можно также создать, нажав соответствующую кнопку на панели инструментов, с помощью меню File (Файл) или соответствующего сочетания клавиш.

Дополнительные источники информации

Вставка дополнительной информации о функции (стр. 63)

Использование команды Save As... (Сохранить как...) (стр. 47)

12.3 Запись кода функции

Введение

Код функции можно записать в окне редактирования функции. Языком программирования является ANSI-C.

С помощью кода любого проекта или стандартной функции можно вызывать другие функции. Вызываемой функцией может быть проект, стандартная, внутренняя функция или функция DLL. Чтобы вызываемая функция была объявлена для вызывающей функции, в качестве первой строки кода вызывающей функции добавляется строка `#include «apdefap.h»`.

В окне навигации в Internal Functions (Внутренние функции) библиотека функций C доступна как «с_bib».

Первая строка кода содержит данные о типе возвращаемого значения и имени по умолчанию новой функции. Параметры можно перенести, если ввести их в следующих скобках.


Введите код функции между фигурными скобками.

Процедура

1. Для открытия функции в окне редактирования дважды щелкните ее в окне навигации.
2. Установите курсор в место, с которого необходимо начать запись.
3. Введите необходимый код.

Альтернативная операция

Функцию можно также открыть следующим образом:

В окне навигации откройте контекстное меню для необходимого макроса и щелкните **Open** (Открыть) или **File\Open...** (Файл/Открыть...). Можно также нажать кнопку  на стандартной панели или использовать соответствующую комбинацию клавиш.

Примечание

Для локальных тегов (теги, определенные в фигурных скобках кода функции) доступно максимум 32 Кб памяти.

Дополнительные источники информации

Использование стандартных функций и функций проекта (стр. 62)

Использование внутренних функций (стр. 61)

Редактирование функций с помощью мыши (стр. 42)

Редактирование функций с помощью клавиатуры (стр. 41)

Работа в окне редактирования (стр. 40)

12.4 Использование внутренних функций

Введение

Можно использовать любую из внутренних функций в качестве части используемого кода функции. Внутренние функции можно найти в окне навигации в группе Internal Functions (Внутренние функции).

Если для добавления функции использовалось диалоговое окно Assigning Parameters (Назначение параметров), тип возвращаемого значения отображается в комментариях функции.

Процедура

1. Поместите курсор в место, куда необходимо вставить внутреннюю функцию.
2. В окне навигации откройте контекстное меню для добавления внутренней функции.
3. Выберите Assigning Parameters (Назначение параметров). Откроется диалоговое окно Assigning Parameters (Назначение параметров).
В этом диалоговом окне у каждого параметра есть своя строка. В столбце Value (Значение) введите значение соответствующего текущего параметра.
4. В столбце Value (Значение) введите текущее значение для каждого необходимого параметра.
Значение можно ввести напрямую с помощью клавиатуры или можно открыть меню в столбце Value (Значение) (щелкнуть один раз, а затем нажать отображаемую кнопку). В меню можно открыть диалоговое окно выбора тегов, кадров или графических объектов.
5. Подтвердите ввод, нажав ОК. Заданные функции можно вставить в окне редактирования в местоположении курсора.

Альтернативная операция

Можно также открыть диалоговое окно Assigning Parameters (Назначение параметров) для внутренней функции, дважды щелкнув функцию, которую требуется добавить.

Примечание

Если закрыть диалоговое окно Assigning Parameters (Назначение параметров) нажатием кнопки ОК, не вводя текущее значение параметра, внутренняя функция вставляется со своими формальными параметрами. Параметры можно позже настроить в окне редактирования.

Вместо использования диалогового окна Assigning Parameters (Назначение параметров) для ввода функции можно также использовать клавиатуру.

12.5 Использование стандартных функций и функций проекта

Введение

Можно использовать любую функцию проекта или стандартную функции как часть кода функции, если перед этим добавить в заголовок строку `#include «ardefap.h»`. Функции проекта находятся в окне навигации в группе Project Functions (Функции проекта). Стандартные функции проекта можно найти в окне навигации в группе Standard Functions (Стандартные функции).

Функции проекта можно ввести в заголовочном файле `Ap_pbib.h`, а стандартные функции можно ввести в заголовочном файле `Ap_glob.h`. Эти данные записываются системой. Заголовочный файл `Ap_glob.h` интегрируется в заголовочный файл `Ap_pbib.h`. Сам заголовочный файл `Ap_pbib.h` связан с заголовочным файлом `Ardefap.h`. Все функции проекта и стандартные функции объявляются в заголовке файла `Ardefap.h`.

Для сообщения компилятору о добавлении функций проекта или стандартных функций добавьте строку `#include «ardefap.h»` в качестве первой строки кода функции.

Если для добавления функции использовалось диалоговое окно Assigning Parameters (Назначение параметров), тип возвращаемого значения отображается в комментариях функций.

Процедура

1. Поместите курсор в место, куда необходимо вставить функцию проекта или стандартную функцию.
2. В окне навигации откройте меню быстрого вызова для функции, которую требуется добавить.
3. Выберите Assigning Parameters (Назначение параметров). Откроется диалоговое окно Assigning Parameters (Назначение параметров).
В этом диалоговом окне у каждого параметра есть своя строка. В столбце Value (Значение) введите значение соответствующего текущего параметра.
4. В столбце Value (Значение) введите текущее значение для каждого необходимого параметра.
Значение можно ввести напрямую с помощью клавиатуры или можно открыть меню в столбце Value (Значение) (щелкнуть один раз, а затем нажать отображаемую кнопку). В меню можно открыть диалоговое окно выбора тегов, кадров или графических объектов.
5. Подтвердите настройки, нажав ОК.

Примечание

Если для функции не требуется параметра, он добавляется к коду функции напрямую, без открытия диалогового окна Assigning Parameters (Назначение параметров).

Если закрыть диалоговое окно Assigning Parameters (Назначение параметров) нажатием кнопки ОК, не вводя текущее значение параметра, внутренняя функция вставляется со своими формальными параметрами. Параметры можно позже настроить в окне редактирования.

12.6 Вставка дополнительной информации о функции

Введение


Для каждой функции можно назначить дополнительную информацию.

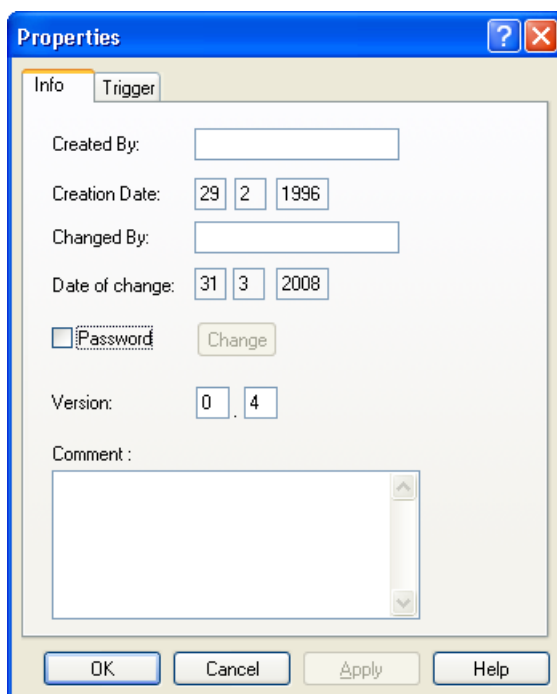
При создании нового макроса дата создания автоматически добавляется к информации о функции, и изменить ее нельзя. Кроме того, номеру присваивается номер версии 1.0. Номера версии можно назначить по отдельности при редактировании функции. При изменении и сохранении функции текущая дата изменения вводится автоматически и ее нельзя изменить. Можно использовать это диалоговое окно для назначения пароля для защиты функции от несанкционированного просмотра и изменения.

Требования

Функция, к которой относится информация, должна быть открыта в окне редактирования.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Выберите необходимые элементы.



3. Подтвердите настройки, нажав ОК.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом:
Откройте меню Edit (Правка) и выберите пункт Info (Информация) или используйте соответствующее сочетание клавиш.

Дополнительные источники информации

Защита функции от несанкционированного доступа (стр. 64)

Настройка разных представлений (стр. 46)

Работа с панелями инструментов (стр. 43)

12.7 Защита функции от несанкционированного доступа


Введение

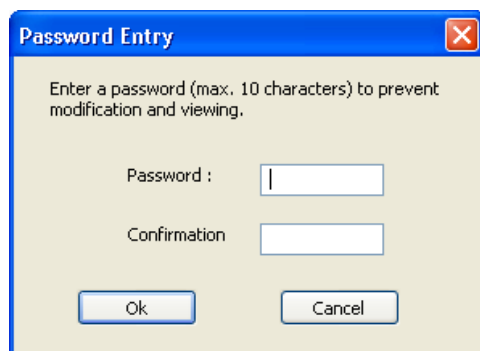
Функции можно защитить паролем от несанкционированного чтения и записи. Пароль является частью информации о функции.

Требования

В окне редактирования должны быть открыты функции, которые требуется компилировать.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Установите флажок Password (Пароль).
3. Нажмите кнопку Change (Изменить).



4. Заполните поле Password (Пароль).
5. Снова введите пароль в поле Confirmation (Подтверждение).
6. Подтвердите настройки, нажав ОК.

7. Нажмите кнопку ОК, чтобы закрыть диалоговое окно.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом: Откройте меню Edit (Правка) и выберите пункт Info (Информация) или используйте соответствующее сочетание клавиш.

Примечание

Функцию, защищенную паролем, можно открыть в окне редактирования только при условии ввода верного пароля.

Для отключения защиты паролем снимите флажок Password (Пароль).

Дополнительные источники информации

Настройка разных представлений (стр. 46)

Работа с панелями инструментов (стр. 43)

12.8 Компилирование и сохранение функции

Введение


Для того чтобы использовать функцию, сначала ее необходимо скомпилировать. Функцию можно скомпилировать только в активном окне редактирования.


Ошибки при компиляции отображаются в нижней части окна. Каждое сообщение отображается в отдельной строке. Строка включает в себя номер строки в исходном коде, где произошла ошибка, закодированный шестнадцатеричный код ошибки и описание ошибки.

Дважды щелкните строку для выбора строки исходного кода, где произошла ошибка.

Рекомендуется проверить первое сообщение в списке ошибок, поскольку последующие ошибки могут стать результатом первой ошибки. Если исправлена первая ошибка, то остальные могут исчезнуть после следующей компиляции.

Чтобы применить изменения, функцию необходимо сохранить.



При сохранении функции, которая не была скомпилирована должным образом, в окне навигации отображается значок .

При сохранении функции, которая была скомпилирована с ошибками, в окне навигации отображается значок .

Требования

В окне редактирования должны быть открыты функции, которые требуется компилировать.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования.
2. Проверьте сообщения компилятора в нижней части окна.
3. Если присутствуют сообщения об ошибках, код функции следует исправить. После этого выполните шаг 1 в этой таблице.
4. Если компилятор выводит предупреждения, возможно, необходимо исправить исходный код функции. После исправления кода выполните шаг 1 в этой таблице, в противном случае перейдите к шагу 6.
5. Нажмите кнопку  на стандартной панели инструментов.

Альтернативная операция

Процесс компиляции можно также выполнить следующими способами:

Выберите Compile (Компилировать) в меню Edit (Правка); выберите пункт Compile (Компилировать) в контекстном меню окна редактирования; или используйте соответствующее сочетание клавиш.

Сохранение можно также выполнить следующими способами:

Выберите Save (Сохранить) в меню File (Файл) или используйте соответствующее сочетание клавиш.

Дополнительные источники информации

Поведение макросов в среде исполнения (стр. 97)

12.9 Переименование функции

Введение

Рекомендуется переименовать функцию при ее создании.



Соответственно, изменяется имя функции в окне редактирования. Поскольку при этом изменяется и код, функцию необходимо повторно скомпилировать. Для старого имени функции, которое отображается в окне навигации, назначается префикс "*" (звездочка).

После этого измененную функцию необходимо сохранить, при этом можно изменить путь к файлу и его имя. Старую функцию следует удалить, чтобы избежать скопления устаревших функций.

Примечание

Обратите внимание, что в имени функции можно использовать только определенные символы: символы (за исключением специальных национальных знаков), цифры и подчеркивание.

Процедура

1. Измените имя функции в окне редактирования.
2. Нажмите кнопку  на панели инструментов редактирования. Выполняется компиляция функции.
3. Нажмите кнопку  на стандартной панели инструментов для сохранения функции
4. При необходимости введите другой путь (или) имя файла.
5. Подтвердите ввод значений, нажав кнопку Save (Сохранить).

Альтернативная операция

Процесс компиляции можно также выполнить следующими способами:

Выберите Compile (Компилировать) в меню Edit (Правка); выберите пункт Compile (Компилировать) в контекстном меню окна редактирования; или используйте соответствующее сочетание клавиш.

Сохранение можно также выполнить следующими способами:

Выберите Save (Сохранить) в меню File (Файл) или используйте соответствующее сочетание клавиш.


12.10 Использование функций из других источников

Введение

В текущем проекте можно также использовать функции из других проектов WinCC и стандартные функции из других систем WinCC. Для этого их необходимо перенести в текущий проект.

Кроме разного местоположения в файловой системе, где они сохранены, различия в процедурах между функциями проекта или стандартных функций отсутствуют.

Процедура

1. Скопируйте функции. Функции проекта следует скопировать в каталог library проекта WinCC. Стандартные функции следует скопировать в подкаталог \arlib каталога WinCC. Содержимое окна навигации обновляется автоматически.
2. Нажмите кнопку  на стандартной панели инструментов. При повторном создании заголовка скопированные функции регистрируются, чтобы их можно было использовать в текущем проекте.

Альтернативная операция

Процесс создания можно также запустить следующим способом:

Откройте меню Options (Параметры) и выберите Regenerate Header (Повторно создать заголовок) или воспользуйтесь соответствующим сочетанием клавиш.

Примечание

В случае переустановки или обновления WinCC измененные стандартные функции удаляются или заменяются неотредактированными стандартными функциями.

Если WinCC находится в среде исполнения, повторное создание заголовков не повлияет на систему среды исполнения.

Создание и редактирование макросов

13.1 Создание и редактирование макросов

Введение

Между глобальными и локальными макросами есть системные различия. В проекте клиент/сервер глобальные макросы выполняются на всех компьютерах в проекте, а локальные выполняются только на компьютерах, для которых они назначены.

Глобальный макрос можно, например, использовать для выполнения расчета на всех компьютерах проекта.

Примером использования локального макроса может служить вывод файла журнала на сервер.

Для обоих типов макросов используются одинаковые процессы создания и редактирования.

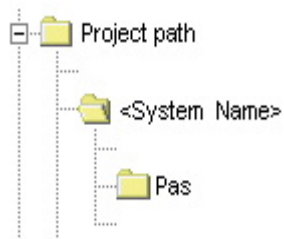
Различие между макросами и функциями

- Макросы в отличие от функций могут иметь триггеры. Это значит, что функция сама не может быть выполнена в среде исполнения.
- Макросы можно экспортировать и импортировать.
- Для макросов можно назначать разрешения. Эти разрешения относятся к рабочим параметрам окна устранения неисправностей в Global Script Runtime (Среда исполнения Global Script).
- У макросов нет параметров.

Поиск макросов

В окне навигации Global Script можно создавать новые макросы и искать существующие.

Макросы хранятся в файловой системе в следующих папках:



Локальные макросы



Глобальные макросы

Редактирование и компиляция макросов

Редактирование и компиляция макросов выполняется в отдельном окне редактирования. В окне редактирования отображаются сообщения компилятора после выполнения компиляции. Это могут быть предупреждения или сообщения об ошибках. В каждом из случаев выводится сводка о количестве предупреждений и сообщений об ошибках.

Отображение макросов



Если в макросе отсутствуют синтаксические ошибки, он отображается в окне навигации Global Script с вышеуказанным значком слева.



Если в макросе отсутствуют синтаксические ошибки и триггер, он отображается в окне навигации Global Script с вышеуказанным значком слева.



Если в макросе отсутствуют синтаксические ошибки и имеется триггер, он отображается в окне навигации Global Script с вышеуказанным значком слева.

При создании макросов обратите внимание на следующее:

Функция CrossReference WinCC позволяет создавать перекрестные ссылки. При создании списка перекрестных ссылок, чтобы теги и кадры распознавались при вызове функций, которые содержатся в макросах, необходимо учитывать правила кодировки, описанные ниже.

Переименование макросов

Макросы всегда отображаются в окне навигации со своими именами файлов. Переименование макроса означает переименование файла, содержащего код макроса.

Поведение системы при изменении, удалении и сохранении макросов в среде исполнения

При сохранении локального макроса в среде исполнения сбрасываются все локальные и глобальные макросы на компьютере, которому принадлежит локальный макрос.

При сохранении глобального макроса в среде исполнения сбрасываются все локальные и глобальные макросы для всего проекта и впоследствии на всех компьютерах.

При таком сбросе может произойти повторная инициализация, например, тегов и времени, которые используются в качестве триггеров для макросов и иницируют их на данном этапе.

Реинициализация выполняется также для статических тегов, которые используются в сброшенных макросах.

Возможные причины невыполнения макроса в среде исполнения

Макрос может быть не выполнен в среде исполнения по одной из следующих причин:

- У макроса нет триггера
- Макрос не был скомпилирован
- В списке запуска проекта отсутствует Global Script Runtime (Среда исполнения Global Script)

Примечание

Перед созданием макроса проверьте, применимы ли соответствующие функции на устройстве автоматизации.

Дополнительные источники информации

Защита макроса от несанкционированного доступа (стр. 79)

Использование макросов из других источников (стр. 95)

Переименование макроса (стр. 94)

Импорт макроса (стр. 93)

Экспорт макроса (стр. 92)

Назначение разрешений (стр. 91)

- Триггеры (стр. 81)
- Компилирование и сохранение макроса (стр. 80)
- Вставка дополнительной информации о макросе (стр. 77)
- Редактирование макросов (стр. 76)
- Создание нового макроса (стр. 75)
- Правило кодирования WinCC (стр. 74)
- Добавление Global Script Runtime (Среда исполнения Global Script) в список запуска проекта (стр. 27)
- Характеристики глобальных макросов (стр. 23)
- Характеристики локальных макросов (стр. 19)

13.2 Правило кодирования WinCC

Правила кодирования для использования функции перекрестных ссылок

Функция CrossReference WinCC позволяет создавать перекрестные ссылки. Чтобы обеспечить распознавание программным обеспечением тегов и кадров, которые используются в вызовах функций, осуществляемых в макросах, необходимо соблюдать правила кодирования, представленные ниже.

Код макроса начинается с двух разделов. В первом разделе необходимо объявить все используемые теги; во втором разделе – все используемые имена кадров.

Не добавляйте никаких дополнительных операторов в эти разделы.

Оба раздела уже присутствуют в форме комментариев при создании макроса:

```
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagname"
// next TagID : 1
// WINCC:TAGNAME_SECTION_END

// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
// WINCC:PICNAME_SECTION_END
```

Разделы можно расширить следующим образом:

```
// WINCC:TAGNAME_SECTION_START
// syntax: #define TagNameInAction "DMTagname"
// next TagID : 1
#define ApcTagName1 "TagName1"
// WINCC:TAGNAME_SECTION_END
```

```
// WINCC:PICNAME_SECTION_START
// syntax: #define PicNameInAction "PictureName"
// next PicID : 1
#define ApcPicName1 "PicName1"
#define ApcPicName2 "PicName2"
#define ApcPicName3 "PicName3"
// WINCC:PICNAME_SECTION_END
```

После этого должны быть обработаны вызовы функций для чтения и записи тегов и использование имен кадров с помощью определенных имен:

```
GetTagDWord (ApcTagName1) ;
OpenPicture (ApcPicName1) ;
SetPictureName (ApcPicName2, "PictureWindow1", ApcPicName3) ;
```

13.3 Создание нового макроса

Введение

В проекте клиент/сервер глобальные макросы выполняются на всех компьютерах в проекте, а локальные выполняются только на компьютерах, для которых они назначены.

Для глобальных и локальных макросов используется одна процедура. При указании в окне навигации местоположения, в котором сохранен макрос, определяется его тип (глобальный или локальный).

В Global Script для нового макроса назначается имя по умолчанию.

Только что созданный макрос уже содержит оператор `#include «apdefap.h»`. Следовательно, все функции уже зарегистрированы в макросе. Имя макроса располагается на третьей строке. Первые три строки нельзя удалить или изменить. Это означает, что каждую функцию можно вызвать из любого макроса, не прибегая к специальным мерам. Кроме того, у каждого макроса есть возвращаемое значение типа `int`, и для него уже установлено значение 0.

Возвращаемое значение макроса можно использовать в GSC Runtime (Среда исполнения макросов) в целях диагностики.

Код макроса начинается с основы кода в форме комментариев. Если эта основа заполнена в соответствии с правилами кодирования, имена тегов и кадров распознаются функцией `CrossReference`.

Процедура

1. В окне навигации откройте контекстное меню для необходимого типа макроса.
2. Выберите **New (Создать)**.

Альтернативная операция

Новый макрос можно также создать нажатием соответствующей кнопки на панели инструментов, с помощью меню File (Файл) или с помощью соответствующего сочетания клавиш.

Дополнительные источники информации

GSC Runtime (Среда исполнения макросов) (стр. 98)

Правило кодирования WinCC (стр. 74)

13.4 Редактирование макросов

Введение

Макрос можно отредактировать в отдельном окне редактирования точно так же, как и функцию. Только первые три строки недоступны для редактирования.

Макрос должен иметь возвращаемое значение. Возвращаемое значение имеет тип int и по умолчанию имеет значение 0. Возвращаемое значение макроса можно изменять и использовать в GSC Runtime (Среда исполнения макросов) в целях диагностики. Тип возвращаемого значения изменить нельзя.


Для выполнения макроса в среде исполнения он должен иметь триггер.

Процедура

1. Для открытия макроса в окне редактирования дважды щелкните его в окне навигации.
2. Отредактируйте код макроса.

Альтернативная операция

Макрос можно также открыть следующим образом:

В окне навигации откройте контекстное меню для необходимого макроса и выберите Open (Открыть) или File\Open... (Файл/Открыть...). Можно также нажать кнопку  на стандартной панели или использовать соответствующую комбинацию клавиш.

Примечание

Для локальных тегов (теги, определенные в фигурных скобках кода макроса) доступно максимум 32 Кб памяти.

Дополнительные источники информации

GSC Runtime (Среда исполнения макросов) (стр. 98)

Запись кода функции (стр. 60)

13.5 Вставка дополнительной информации о макросе

Введение

Для каждого макроса можно назначить дополнительную информацию.


При создании нового макроса в информацию о макросе автоматически добавляется дата создания, которую невозможно изменить. Для макроса назначается номер версии 1.0. Номера версий можно назначать отдельно при редактировании макроса. При изменении или сохранении макроса автоматически добавляется текущая дата изменения, которую невозможно изменить.

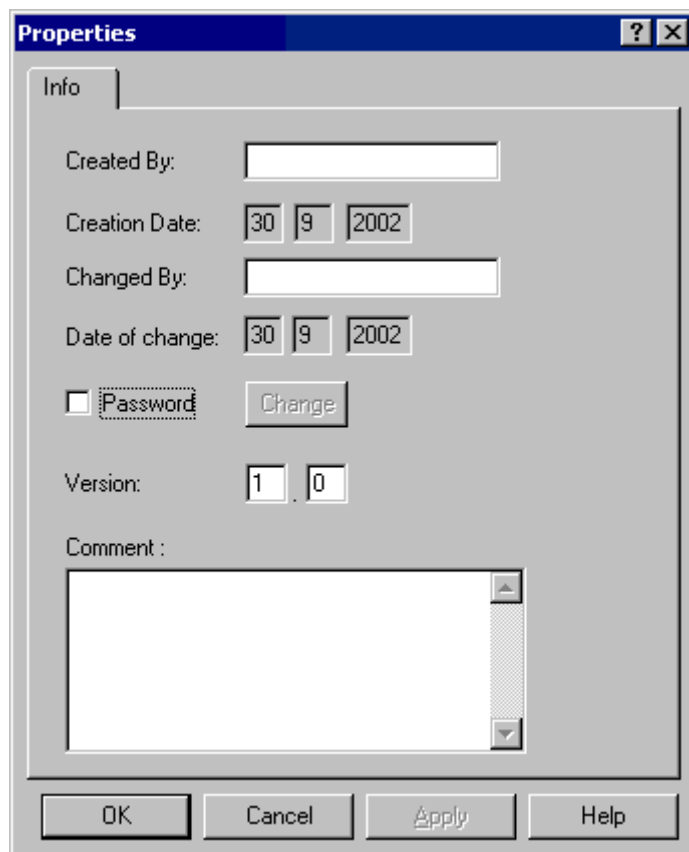
В этом диалоговом окне можно назначить пароль для защиты макроса от несанкционированного чтения и записи.

Требования

Макрос, к которому относится информация, должен быть открыт в окне редактирования.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Выберите необходимые элементы.



3. Подтвердите настройки, нажав ОК.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом:
Откройте меню Edit (Правка) и выберите пункт Info (Информация) или используйте соответствующее сочетание клавиш.

Дополнительные источники информации

- Защита макроса от несанкционированного доступа (стр. 79)
- Настройка разных представлений (стр. 46)
- Работа с панелями инструментов (стр. 43)

13.6 Защита макроса от несанкционированного доступа


Введение

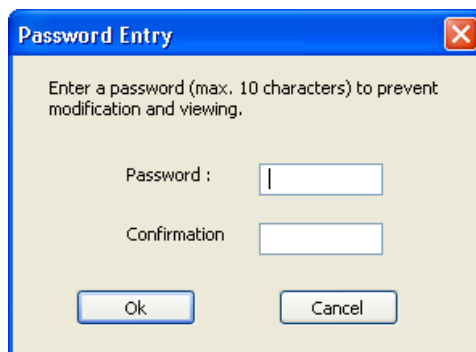
Макросы можно защитить паролем от несанкционированного чтения и записи. Пароль является частью информации о макросе.

Требования

В окне редактирования должны быть открыты макросы, которые требуется компилировать.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Установите флажок Password (Пароль).
3. Нажмите кнопку Change (Изменить).



4. Заполните поле Password (Пароль).
5. Снова введите пароль в поле Confirmation (Подтверждение).
6. Подтвердите настройки, нажав ОК.
7. Нажмите кнопку ОК, чтобы закрыть диалоговое окно.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом: Откройте меню Edit (Правка) и выберите пункт Info (Информация) или используйте соответствующее сочетание клавиш.

Примечание

Макрос, защищенный паролем, можно открыть в окне редактирования только при условии ввода верного пароля.

Для отключения защиты паролем снимите флажок Password (Пароль).

Дополнительные источники информации

Настройка разных представлений (стр. 46)

Работа с панелями инструментов (стр. 43)

13.7 Компилирование и сохранение макроса

Введение

Для использования макроса сначала его необходимо скомпилировать. Макрос можно скомпилировать только в активном окне редактирования.

Ошибки компиляции отображаются в нижней части окна. Каждое сообщение отображается в отдельной строке. Строка включает в себя номер строки в исходном коде, где произошла ошибка, закодированный шестнадцатеричный код ошибки и описание ошибки.



Дважды щелкните такую строку для просмотра строки в исходном коде, где произошла ошибка.

Рекомендуется проверить первое сообщение в списке ошибок, поскольку последующие ошибки могут стать результатом первой ошибки. Если исправлена первая ошибка, то остальные могут исчезнуть после следующей компиляции.

Требования

В окне редактирования должны быть открыты макросы, которые требуется компилировать.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования.
2. Проверьте сообщения компилятора в нижней части окна.
3. Если присутствуют сообщения об ошибках, исходный код макроса следует исправить. После этого выполните шаг 1 в этой таблице.
4. Если компилятор выводит предупреждения, возможно, необходимо исправить исходный код макроса. После исправления кода выполните шаг 1 в этой таблице, в противном случае перейдите к шагу 6.
5. Нажмите кнопку  на стандартной панели инструментов.

Альтернативная операция

Процесс компиляции можно также выполнить следующими способами:

Выберите Compile (Компилировать) в меню Edit (Правка); выберите пункт Compile (Компилировать) в контекстном меню окна редактирования; или используйте соответствующее сочетание клавиш.

Сохранение можно также выполнить следующими способами:

Выберите Save (Сохранить) в меню File (Файл) или используйте соответствующее сочетание клавиш.

Отображение макросов



Если в макросе отсутствуют синтаксические ошибки, он отображается в окне навигации Global Script с вышеуказанным значком слева.



Если в макросе отсутствуют синтаксические ошибки и триггер, он отображается в окне навигации Global Script с вышеуказанным значком слева.



Если в макросе отсутствуют синтаксические ошибки и имеется триггер, он отображается в окне навигации Global Script с вышеуказанным значком слева.

Дополнительные источники информации

Поведение макросов в среде исполнения (стр. 97)

13.8 Триггеры

13.8.1 Триггеры

Определение и использование триггеров

Триггеры используются для выполнения макросов в среде исполнения. Поэтому триггер связан с макросом, образуя иницилирующее событие для вызова макроса. Макросы без триггеров не выполняются.

Типы триггеров

Доступны следующие типы триггеров:

Апериодические триггеры

Включают в себя заданные дату и время. Макрос с таким триггером выполняется один раз в заданный день и время.

Циклические триггеры

Включают в себя заданный интервал времени и начальную точку. Существуют следующие типы циклических триггеров:

- Default cycle (Стандартный цикл). Начало первого интервала совпадает с запуском среды исполнения. Длина интервалов определяется циклом.
- Hourly (Раз в час). Начало интервала указывается в минутах и секундах. Длина интервала: час.
- Daily (Раз в день). Начало интервала намечено на определенное время (часы, минуты и секунды). Длина интервала: день.
- Weekly (Раз в неделю). В качестве начала интервала указывается день недели (Monday (понедельник), Tuesday (вторник) и т.д.) и время. Длина интервала: неделя.
- Monthly (Раз в месяц). В качестве начала интервала указывается день и время. Длина интервала: месяц.
- Annually (Раз в год). В качестве начала интервала указывается день, месяц и время. Длина интервала: год.

Тег-триггеры

Включают в себя спецификацию одного или нескольких тегов. Макрос, связанный с таким триггером, выполняется каждый раз при обнаружении изменения значения одного из таких тегов.

Способ запроса значений тегов можно настроить для каждого тега. Можно выбрать циклический опрос с указанным периодом или ответное действие при обнаружении системой изменения значения тега.

В зависимости от выбранного способа запроса изменение значения тега может быть не зафиксировано системой. В этом случае макрос не выполняется.

Действие триггеров на макросы

Если макрос связан только с одним триггером, то этот макрос выполняется сразу после того, как происходит инициирующее событие.

Однако макрос может быть связан с несколькими триггерами, такими как циклический триггер и тег-триггер. В этом случае макрос выполняется при возникновении одного из инициирующих событий. При возникновении двух событий одновременно макрос последовательно выполняется дважды. Если два тег-триггера срабатывают одновременно, макрос выполняется только один раз.

Обработка макроса должна быть завершена до вызова другого макроса, поскольку в противном случае может возникнуть переполнение очереди.

Подсказка. Если макрос не требуется выполнять при каждом возникновении события, то для макроса можно указать условие, контролирующее дальнейшее его выполнение в зависимости от события. Если больше не требуется выполнять макрос, его выполнение можно завершить возвратом значения <value>.

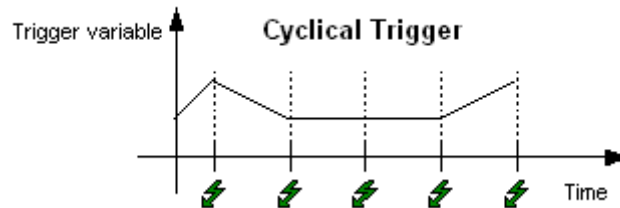
Правила выбора триггеров

В зависимости от системы нельзя гарантировать, что макрос с циклическим триггером будет выполняться точно в указанное время. Если такое требование все же

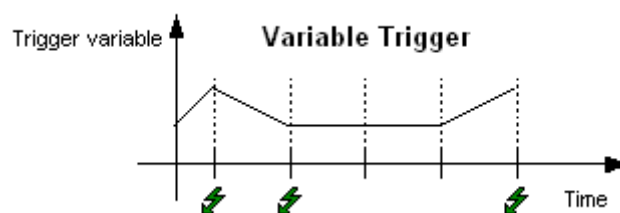
существует, то задачу (например, проверку и т.д.) следует реализовать на устройстве автоматизации.

При обработке тег-триггеры имеют приоритет над циклическими триггерами.

- При использовании циклических триггеров макросы всегда выполняются при возникновении инициирующего события, например каждые 20 секунд. (⚡ = выполнение макроса)



- При использовании тег-триггера макрос будет выполняться только при изменении значения тега триггера. Это делается для снижения нагрузки на систему (⚡ = выполнение макроса).



При запуске макроса значения тегов, содержащихся в триггере, уже объявлены. Можно использовать вызов GetTag() для получения значения напрямую. Обработка тегов, содержащихся в триггерах, выполняется быстрее, чем тегов, которые не содержатся в триггерах и чьи значения необходимо получать с помощью запросов GetTag().

Отображение макросов



Если в макросе отсутствуют синтаксические ошибки, он отображается в окне навигации Global Script с вышеуказанным значком слева.



Если в макросе отсутствуют синтаксические ошибки и триггер, он отображается в окне навигации Global Script с вышеуказанным значком слева.



Если в макросе отсутствуют синтаксические ошибки и имеется триггер, он отображается в окне навигации Global Script с вышеуказанным значком слева.

Дополнительные источники информации

Удаление триггера (стр. 90)

Изменение триггера (стр. 88)

Добавление нового триггера типа Tag (Тег) (стр. 86)

Добавление нового триггера типа Timer (Таймер) (стр. 84)

13.8.2 Добавление нового триггера типа Timer (Таймер)

Введение

Триггеры используются для выполнения макросов в среде исполнения. Поэтому триггер связан с макросом, образуя инициирующее событие для вызова макроса. Макросы без триггеров не выполняются.

Триггеры типа Timer (Таймер) могут быть циклическими или аperiodическими.

Аperiodические триггеры включают в себя заданные дату и время. Макрос с таким триггером выполняется один раз в заданный день и время.


Циклические триггеры включают в себя заданный интервал времени и начальную точку. Существуют следующие типы циклических триггеров:

- Default cycle (Стандартный цикл). Начало первого интервала времени совпадает с запуском системы среды исполнения. Длина интервала определяется циклом.
- Hourly (Раз в час). Начало интервала указывается в минутах и секундах. Длина интервала: час.
- Daily (Раз в день). Начало интервала намечено на определенное время (часы, минуты и секунды). Длина интервала: день.
- Weekly (Раз в неделю). В качестве начала интервала указывается день недели (Monday (понедельник), Tuesday (вторник) и т.д.) и время. Длина интервала: неделя.
- Monthly (Раз в месяц). В качестве начала интервала указывается день и время. Длина интервала: месяц.
- Annually (Раз в год). В качестве начала интервала указывается день, месяц и время. Длина интервала: год.

Требования

В активном окне редактирования должен быть открыт макрос, который требуется связать с триггером.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Перейдите на вкладку Trigger (Триггер).
3. Выберите тип триггера Timer (Таймер) и нажмите кнопку Add (Добавить).
4. Выберите Single (Однократно) для добавления нециклического триггера или Cycle (Циклический) для добавления циклического триггера.



5. В диалоговом окне выполните необходимые настройки.
6. Подтвердите настройки, нажав ОК.
7. Закройте диалоговое окно Properties (Свойства), нажав кнопку ОК.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом: Выберите Info (Информация) в меню Edit (Правка), выберите параметр Info / Trigger (Информация/триггер) в контекстном меню окна редактирования или используйте соответствующее сочетание клавиш.

13.8.3 Добавление нового триггера типа Tag (Тег)

Введение

Триггеры используются для выполнения макросов в среде исполнения. Поэтому триггер связан с макросом, образуя инициирующее событие для вызова макроса. Макросы без триггеров не выполняются.

Тег-триггеры включают в себя один или несколько заданных тегов. Макрос, связанный с таким триггером, выполняется каждый раз при обнаружении изменения значения одного из таких тегов.


Способ запроса значений тегов можно настроить для каждого тега. Можно выбрать циклический опрос с указанным периодом или ответное действие при обнаружении системой изменения значения тега.

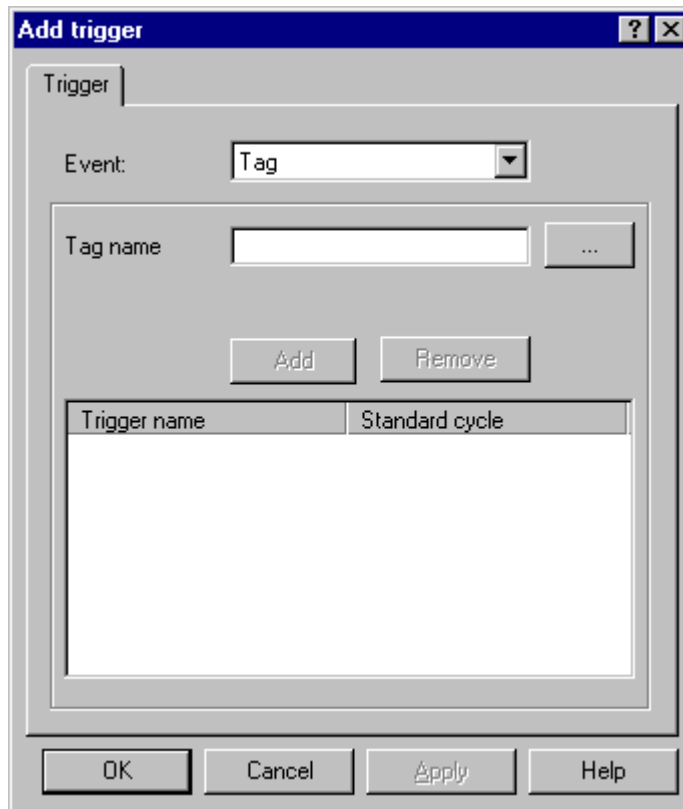
В зависимости от выбранного способа запроса изменение значения тега может быть не зафиксировано системой. В этом случае макрос не выполняется.


Требования

В активном окне редактирования должен быть открыт макрос, который требуется связать с триггером.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Перейдите на вкладку Trigger (Триггер).
3. Выберите тип триггера Tag (Тег) и нажмите кнопку Add (Добавить). Откроется диалоговое окно Add Tags (Добавление тегов).



4. Нажмите кнопку  для открытия диалогового окна выбора тегов, выберите тег и подтвердите выбор, нажав кнопку ОК.
5. В диалоговом окне Add Trigger (Добавление триггера) откройте контекстное меню в столбце Standard cycle (Стандартный цикл) и затем выберите необходимый цикл мониторинга. Параметр After Every Change (После каждого изменения) означает непрерывный мониторинг.
6. При необходимости добавления тегов повторите шаги 4 и 5.
7. Подтвердите настройки, нажав ОК.
8. Закройте диалоговое окно Properties (Свойства), нажав кнопку ОК.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом: Выберите Info (Информация) в меню Edit (Правка), выберите параметр Info / Trigger (Информация/триггер) в контекстном меню окна редактирования или используйте соответствующее сочетание клавиш.

В диалоговом окне Add Trigger (Добавление триггера) можно также ввести имя тега напрямую и вставить тег в столбце Trigger Name (Имя триггера), нажав кнопку Add (Добавить). Однако при этом система не проверяет, существует ли этот тег.

13.8.4 Изменение триггеров


Введение

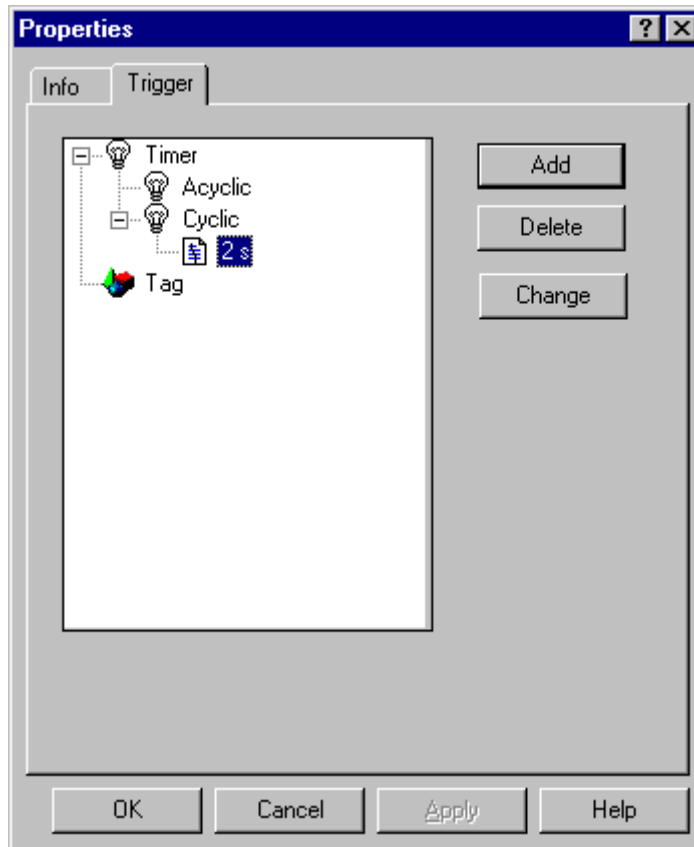
Заданные триггеры можно изменить в любое время. Их также можно изменить в среде исполнения.

Требования

В окне редактирования должен быть открыт соответствующий макрос.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Перейдите на вкладку Trigger (Триггер) и выберите триггер, который необходимо изменить.



3. Нажмите кнопку Change (Изменить), чтобы открыть диалоговое окно Change Trigger (Изменение триггера).
4. Внесите необходимые изменения.
5. Подтвердите настройки, нажав ОК.
6. Закройте диалоговое окно Properties (Свойства), нажав кнопку ОК.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом: Выберите Info (Информация) в меню Edit (Правка), выберите параметр Info / Trigger (Информация/триггер) в контекстном меню окна редактирования или используйте соответствующее сочетание клавиш.

13.8.5 Удаление триггера

Введение


Заданные триггеры можно удалить в любое время. Их также можно удалить в среде исполнения.

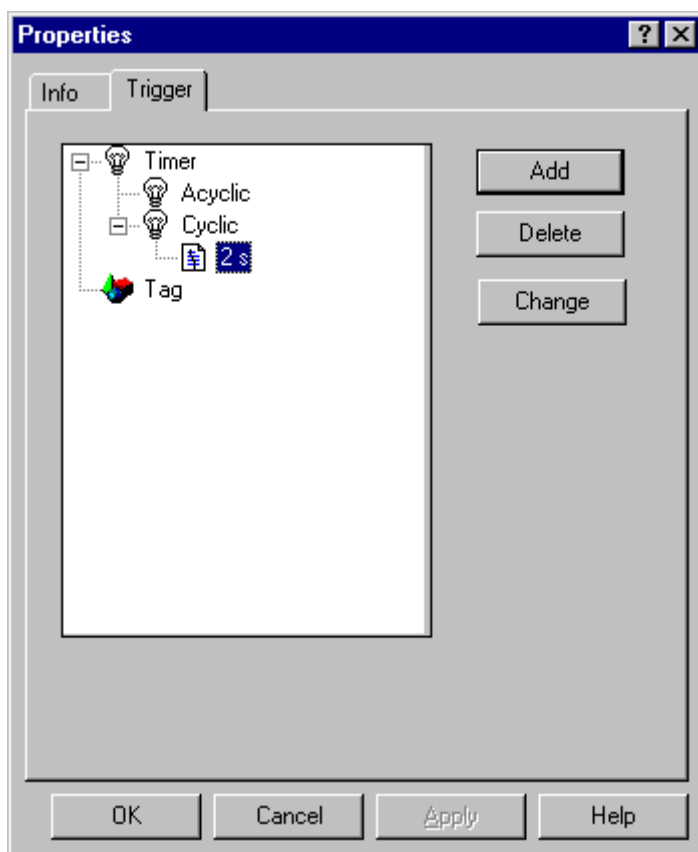
Если триггер удаляется в среде исполнения, удаление вступает в силу только после сохранения макроса.

Требования

В окне редактирования должен быть открыт соответствующий макрос.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Properties (Свойства).
2. Перейдите на вкладку Trigger (Триггер) и выберите триггер, который необходимо удалить.



3. Удалите выбранный триггер, нажав кнопку Delete (Удалить).
4. Закройте диалоговое окно Properties (Свойства), нажав кнопку OK.

Альтернативная операция

Диалоговое окно Properties (Свойства) можно также открыть следующим способом:
Выберите Info (Информация) в меню Edit (Правка), выберите параметр Info / Trigger (Информация/триггер) в контекстном меню окна редактирования или используйте соответствующее сочетание клавиш.

13.9 Назначение разрешений

Введение

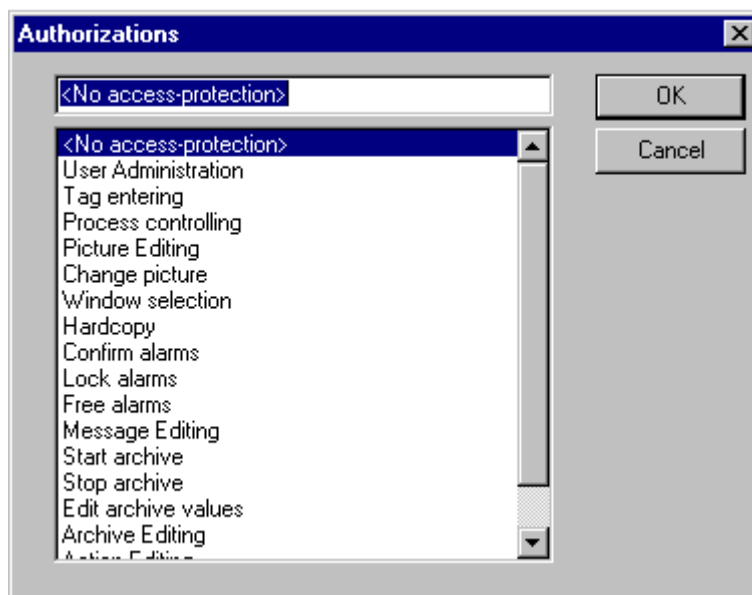
С помощью инструмента диагностики среды исполнения Global Script можно влиять на обработку макросов в среде исполнения. Для каждого макроса можно назначить разрешение. Это разрешение влияет только на работу в окне среды исполнения Global Script.

Требования

В окне редактирования должен быть открыт соответствующий макрос.

Процедура

1. Выберите команду Operator authorization (Разрешение оператора) в меню Edit (Правка).
Откроется диалоговое окно Authorizations (Разрешения).
2. Выберите разрешение.



3. Подтвердите выбор, нажав кнопку OK.

Дополнительные источники информации

GSC Runtime (Среда исполнения макросов) (стр. 98)

13.10 Экспорт макроса


Введение

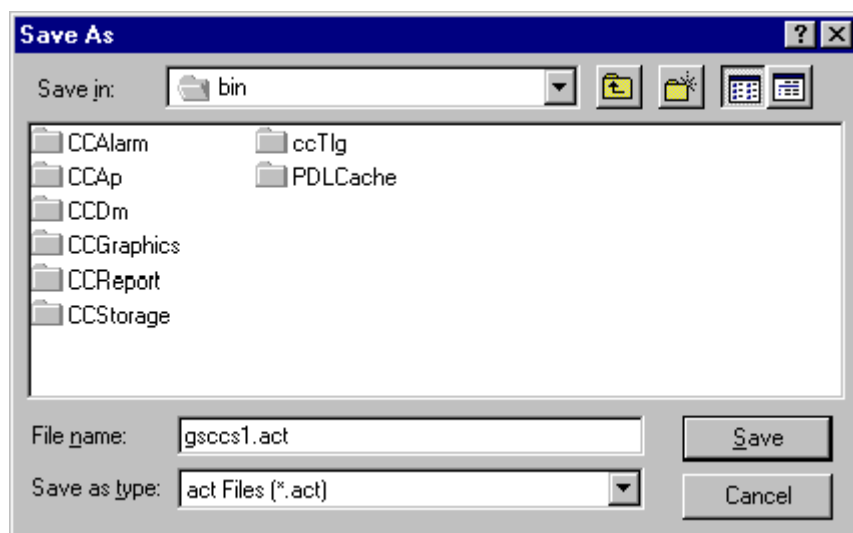
Экспорт и импорт используются для переноса макросов в другие проекты. Триггеры, связанные с макросами, сохраняются в процессе.

Требования

В окне редактирования должен быть открыт макрос, который требуется экспортировать.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Save As (Сохранить как).
2. Выберите путь и имя файла для макроса, который необходимо экспортировать.



3. Закройте диалоговое окно, нажав кнопку Save (Сохранить).

Альтернативная операция

Операцию экспорта можно также запустить следующим образом:

Выберите Export (Экспорт) в меню Edit (Правка), выберите параметр Export (Экспорт) в контекстном меню окна редактирования или используйте соответствующее сочетание клавиш.


13.11 Импорт макроса

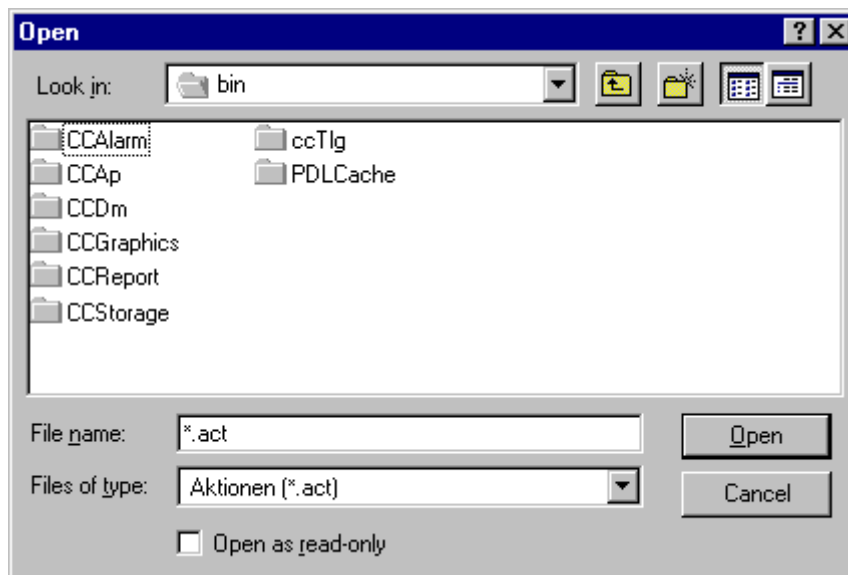
Введение

Экспорт и импорт используются для переноса макросов в другие проекты. Триггеры, связанные с макросами, сохраняются в процессе.

Макрос в активном окне редактирования заменяется импортированным макросом.

Процедура

1. Нажмите кнопку  на панели инструментов редактирования. Откроется диалоговое окно Open (Открыть).
2. Выберите путь и имя файла для макроса, который необходимо импортировать.



3. Закройте диалоговое окно, нажав кнопку Open (Открыть).

Альтернативная операция

Операцию импорта можно также запустить следующим образом:

Выберите Import (Импорт) в меню Edit (Правка), выберите параметр Import (Импорт) в контекстном меню окна редактирования или используйте соответствующее сочетание клавиш.

13.12 Переименование макроса

Введение

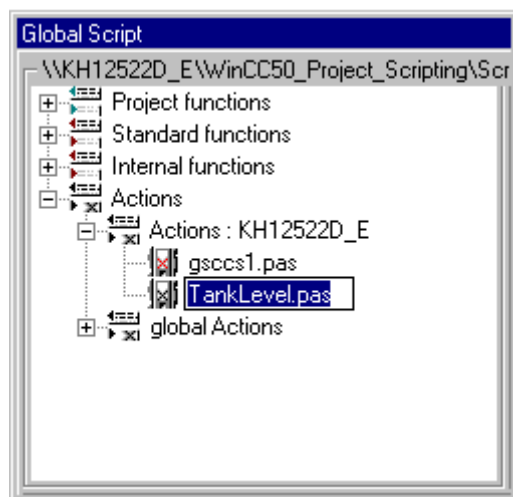
Макрос можно переименовать в любое время. При этом для него назначается другое имя файла.

Требования

В окне редактирования должен быть открыт макрос, который требуется переименовать.

Процедура

1. Откройте контекстное меню макроса, который требуется переименовать.
2. Выберите Rename (Переименовать).
3. Введите новое имя с расширением файла «.pas».



4. Подтвердите новое имя, нажав клавишу <ВВОД>.

Альтернативная операция

Можно также дважды щелкнуть имя макроса для завершения переименования.

Примечание

Если не ввести расширение файла «.pas», имя макроса останется без изменений.

13.13 Использование макросов из других источников

Введение

Макросы из других источников можно использовать в проекте двумя способами:

- Путем импорта экспортированных макросов
- Путем копирования файла с необходимым макросом в соответствующую папку проекта. Папкой для локальных макросов является «<Имя_компьютера>\Pas» в папке проекта. Папкой для глобальных макросов является «\Pas» в папке проекта.

Для отображения скопированных макросов в окне навигации необходимо обновить экран. Для этого можно перезапустить Global Script.

Если среда исполнения запущена, импортированные макросы выполняются только после открытия и последующего сохранения в редакторе Global Script.

Примечание

Макросы могут содержать вызовы стандартных функций и функций проекта. Они также могут содержать вызовы функций проекта и стандартных функций и т.д. Поэтому при импорте макросов из других источников необходимо убедиться, что в текущем проекте имеются все необходимые функции.

Особое внимание следует обратить при импорте макросов с другого компьютера. Поскольку стандартные функции доступны для настройки пользователем, может получиться так, что стандартные функции, вызываемые макросом, будут иметь другое назначение на исходном компьютере, которое отличается от назначения стандартных функций на целевом компьютере.

Дополнительные источники информации

Импорт макроса (стр. 93)

Экспорт макроса (стр. 92)

Поведение макросов в среде исполнения

14.1 Поведение макросов в среде исполнения

Анализ поведения в среде исполнения

В WinCC имеется несколько средств, с помощью которых можно проанализировать поведение макросов в среде исполнения. Это окна приложений GSC Runtime (Среда исполнения макросов) и GSC Diagnose (Диагностика макросов), а также приложение apdiag.exe.

Для использования окон приложений GSC Runtime (Среда исполнения макросов) и GSC Diagnose (Диагностика макросов) их необходимо добавить в кадр процесса. Этим кадром должен быть кадр процесса, который предназначен только для диагностики. Этот кадр затем вызывается в среде исполнения.

Окна приложений используются в следующих целях:

- GSC Runtime (Среда исполнения макросов) обеспечивает информацию о динамическом поведении всех макросов (Global Script), поддерживает однократный запуск макросов, а также запуск и завершение для каждого отдельного макроса и предоставляет точку входа в редактор Global Script при запущенной среде исполнения.
- GSC Diagnose (Диагностика макросов) выводит операторы printf (содержащиеся в макросах) в том порядке, в котором они вызываются. Это также относится к операторам printf в функциях, которые вызываются в макросах. При тщательно продуманном использовании операторов printf, например, для вывода значений тегов, можно следить за потоком макросов и вызываемых функций. Даже состояния ошибки, которые приводят к вызову функции OnErrorExecute, отображаются в окне GSC Diagnose (Диагностика макросов).

Дополнительные источники информации

GSC Diagnose (Диагностика макросов) (стр. 105)

GSC Runtime (Среда исполнения макросов) (стр. 98)

14.2 GSC Runtime (Среда исполнения макросов)

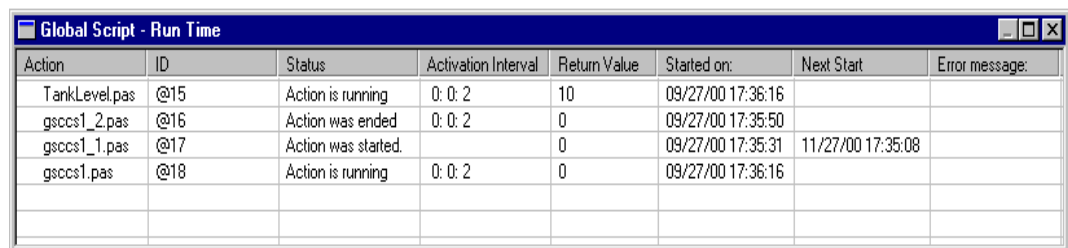
14.2.1 GSC Runtime (Среда исполнения макросов)

Окно GSC Runtime (Среда исполнения макросов)

GSC Runtime (Среда исполнения макросов) – это окно, в котором отображается динамическое поведение всех макросов (Global Script) в среде исполнения. Кроме того, GSC Runtime (Среда исполнения макросов) позволяет влиять на выполнение каждого отдельного макроса и обеспечивает точку входа в редактор Global Script при запущенной среде исполнения.

Выводится следующая информация:

- Action (Макрос): имя макроса:
- ID (Идентификатор): идентификатор макроса. Он используется в рамках системы и, например, выводится функцией OnErrorExecute вместе с описанием ошибки, если в макросе возникла ошибка. В GSC Runtime (Среда исполнения макросов) к имени макроса привязывается этот идентификатор. Связь между идентификатором и именем макроса разрывается при завершении работы среды исполнения или сохранении макроса при запущенной среде исполнения.
- Status (Состояние): текущее состояние макроса. Для получения информации о возможных состояниях см. таблицу ниже.
- Activation Interval (Интервал запуска): время в формате «часы:минуты:секунды», которое прошло между двумя вызовами макроса.
- Return Value (Возвращаемое значение): возвращаемое значение макроса.
- Started On (Запущен): дата и время запуска текущего макроса.
- Next Start (Следующий запуск): дата и время повторного запуска макроса.
- Error Message (Сообщение об ошибке): текст ошибки в случае ее возникновения.



Action	ID	Status	Activation Interval	Return Value	Started on:	Next Start	Error message:
TankLevel.pas	@15	Action is running	0: 0: 2	10	09/27/00 17:36:16		
gscs1_2.pas	@16	Action was ended	0: 0: 2	0	09/27/00 17:35:50		
gscs1_1.pas	@17	Action was started.		0	09/27/00 17:35:31	11/27/00 17:35:08	
gscs1.pas	@18	Action is running	0: 0: 2	0	09/27/00 17:36:16		

Окно GSC Runtime (Среда исполнения макросов)

Состояния макросов

Возможные состояния макросов:

- Action was started (Макрос запущен).
- Action was ended (Макрос завершен).
- Action was stopped (Макрос остановлен).
- Action is running (Макрос выполняется).

- Error during start of action! (Ошибка запуска макроса!)
- Error during execution of action! (Ошибка выполнения макроса!)

Сообщения об ошибках

Возможные сообщения об ошибках:

- No error occurred (Ошибок нет).
- The application is already connected to the script control. No additional connection setup is possible (Приложение уже соединено с менеджером макросов. Установление дополнительного соединения невозможно).
- There is no connection to the script control. Possibly no connection setup took place (Отсутствует соединение с менеджером макросов. Вероятно, соединение не было установлено).
- An error occurred during interprocess communication. The cause of the error is unknown (Ошибка обмена данными между процессами. Причина ошибки неизвестна).
- Undefined error (Неизвестная ошибка).
- The parameter assignment is wrong. Some necessary parameters may be missing (Неверное назначение параметра. Возможно, отсутствуют некоторые необходимые параметры).
- Script control is not started. Check has whether WinCC has been started (Менеджер макросов не запущен. Проверьте, запущена ли WinCC).
- Time-out occurred. Check the connection or increase the monitoring time (Таймаут. Проверьте соединение или увеличьте время мониторинга).
- Script control was terminated (Работа менеджера макросов была прервана).
- The service channel could not be installed (Не удалось установить канал обслуживания).
- An unknown job number was used for the EndAct job (Для задания EndAct был использован неизвестный номер задания).
- The action could not be executed without error. The returned results are invalid (При выполнении макроса возникли ошибки. Возвращаемые результаты недопустимы).
- An error occurred in the server application (В серверном приложении произошла ошибка).
- The maximum number of connections to the script control has been reached (Достигнуто максимальное число соединений с менеджером макросов).
- The transaction is unknown. An attempt was made to terminate a transaction that was not logged on previously (Неизвестная транзакция. Была предпринята попытка завершения не зарегистрированной ранее транзакции).
- A pre-compiled header file cannot be generated from a pre-compiled header file (Не удастся сгенерировать предварительно скомпилированный заголовочный файл с использованием предварительно скомпилированного заголовочного файла).
- There is no access to the action. The module is being used presently (Нет доступа к макросу. Модуль используется в настоящий момент).
- The program is invalid (Недопустимая программа).
- The action is invalid (Недопустимый макрос).
- The script control could not set up the file (Менеджеру макросов не удалось создать файл).
- The script interpreter does not have enough memory (Интерпретатору макросов не хватает памяти).

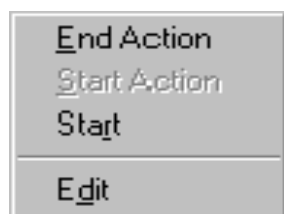
- The file format is invalid for the script control (Формат файла не подходит для менеджера макросов).
- The script control could not open the file (Менеджеру макросов не удалось открыть файл).
- The program is presently locked by the script control. No further access is possible (Программа в данный момент заблокирована менеджером макросов. В доступе отказано).
- The action has already been given to the script control for processing (Макрос уже был передан в менеджер макросов для обработки).
- In this action, a conflict has occurred with another action (Произошел конфликт данного макроса с другим макросом).
- The script control could not find the action (Менеджеру макросов не удалось найти макрос).
- The script control could not find the function (Менеджеру макросов не удалось найти функцию).
- The specified line information is invalid (Указанные данные строки неверны).
- The specified symbol is outside the valid range (Указанный символ лежит вне допустимого диапазона).
- The provided memory is too small for the script interpreter (Выделенной памяти недостаточно для интерпретатора макросов).
- The script interpreter does not recognize the specified type (Интерпретатор макросов не распознает указанный тип).
- The specified symbol was not found (Указанный символ не найден).
- Load project functions (Загрузка функций проекта).
- A stack overflow has occurred in the script interpreter during execution. Further execution of the action is being canceled (При выполнении макроса произошло переполнение стека интерпретатора. Дальнейшее выполнение макроса отменяется).
- Division by 0 occurred during execution of an action. The action is being canceled (При выполнении макроса произошло деление на 0. Выполнение макроса отменяется).
- Within the action, a reference was made during execution to a symbol that does not exist (В макросе во время выполнения была использована ссылка на несуществующий символ).
- Within the action, an attempt was made during execution to access an undefined memory area (В макросе во время выполнения была предпринята попытка доступа к неопределенной области памяти).
- The script interpreter ran into a breakpoint (Интерпретатор макроса дошел до точки останова).
- The script interpreter was advanced in the debugger by one processing step (Интерпретатор макроса был перенесен отладчиком на один шаг обработки вперед).
- The action contains no interpreter code (Макрос не содержит код интерпретатора).
- The action has the wrong data format (Неверный формат данных макроса).
- The return value of the action cannot be represented as a variant (Значение, возвращаемое макросом, не может быть представлено в виде variant).

- There is insufficient memory to execute this operation (Недостаточно памяти для выполнения данной операции).
- An error has occurred within the transaction. For more information, see the AP_ACT_KEYS (Ошибка транзакции. Для получения дополнительной информации см. AP_ACT_KEY).
- An error occurred while executing the action. For more information, see the AP_ACT_KEYS (Ошибка выполнения макроса. Для получения дополнительной информации см. AP_ACT_KEY).
- An error occurred while executing the action. For more information, see the AP_ACT_KEYS (Ошибка выполнения макроса. Для получения дополнительной информации см. AP_ACT_KEY).
- There is no update capability for the existing data format. The action cannot be read (Для существующего формата данных отсутствует возможность обновления. Не удается считать макрос).

Контекстное меню для макросов

Для каждого макроса в контекстном меню доступны следующие функции:

- End Action (Завершить макрос): соответствующий макрос не будет выполняться снова после завершения текущего выполнения.
- Start Action (Запустить макрос): соответствующий макрос выполняется снова при следующем возникновении иницилирующего события.
- Start (Запустить): соответствующий макрос выполняется однократно.
- Edit (Изменить): соответствующий макрос открывается в редакторе Global Script для редактирования. Работа среды исполнения не завершается. При компиляции (при необходимости) и сохранении отредактированного макроса изменения немедленно вступают в силу в среде исполнения.



Контекстное меню

Для каждого отдельного макроса можно указать, будет ли требоваться пароль для открытия всплывающего меню.

Для использования GSC Runtime (Среда исполнения макросов) сначала необходимо добавить окно приложения типа GSC Runtime (Среда исполнения макросов) в кадр процесса. С помощью атрибутов GSC Runtime (Среда исполнения макросов) можно настроить внешний вид окна GSC Runtime (Среда исполнения макросов).

Примечание

При обновлении окна GSC Runtime (Среда исполнения макросов) увеличивается нагрузка на ресурсы системы. Нагрузка на систему зависит от того, сколько макросов отображается в этом окне. Нагрузку на систему можно снизить, уменьшив высоту окна, чтобы отображалось меньше строк.

Дополнительные источники информации

Редактирование макросов (стр. 104)

Атрибуты GSC Runtime (Среда исполнения макросов) (стр. 104)

Помещение GSC Runtime (Среда исполнения макросов) в кадр процесса (стр. 103)

Назначение разрешений (стр. 91)

14.2.2 Помещение GSC Runtime (Среда исполнения макросов) в кадр процесса

Введение

Чтобы использовать GSC Runtime (Среда исполнения макросов), ее необходимо добавить в кадр процесса. Этим кадром процесса может быть существующий кадр или кадр, служащий исключительно в целях диагностики. GSC Runtime (Среда исполнения макросов) не удастся добавить в кадр процесса напрямую, ее следует добавить в качестве приложения в окне приложения. Окно приложения является частью кадра процесса. Описанные здесь действия необходимо выполнить в графическом дизайнера.

Требования

Должен быть запущен графический дизайнер и открыт кадр процесса.

Процедура

1. На палитре объектов выберите Smart Object\Application Window (Интеллектуальный объект\окно приложения).
2. В области отрисовки откройте окно приложения.
3. В диалоговом окне Window Contents (Содержимое окна) выберите Global Script.
4. Подтвердите настройки, нажав ОК.
5. В диалоговом окне Template (Шаблон) выберите GSC Runtime (Среда исполнения макросов).
6. Нажмите кнопку ОК, чтобы подтвердить выбор.

Дополнительные источники информации

Атрибуты GSC Runtime (Среда исполнения макросов) (стр. 104)

14.2.3 Атрибуты GSC Runtime (Среда исполнения макросов)

Макет окна GSC Runtime (Среда исполнения макросов)

GSC Runtime (Среда исполнения макросов) имеет атрибуты, с помощью которых можно настроить вид окна GSC Runtime (Среда исполнения макросов) в среде исполнения. Сюда входят атрибут геометрии и, в частности, следующие атрибуты:

- **Display (Отображение):** с помощью этого атрибута можно отобразить или скрыть окно. Атрибут можно сделать динамическим с помощью имени Visible (Видимый).
- **Sizeable (Разрешить изменение размера):** с помощью этого атрибута можно разрешить или запретить изменение размера окна в среде исполнения.
- **Moveable (Разрешить перемещение):** с помощью этого атрибута можно разрешить или запретить перемещение окна в среде исполнения.
- **Border (Граница):** с помощью этого атрибута можно включить или отключить границу для окна. Если окно будет иметь границу, его ширину и высоту можно будет изменить в среде исполнения.
- **Title (Заголовок):** с помощью этого атрибута можно включить или отключить использование строки заголовка в окне.
- **Can be Maximized (Разрешить разворачивание):** с помощью этого атрибута можно включить или отключить использование кнопки в строке заголовка окна для разворачивания окна в среде исполнения.
- **Can be Closed (Разрешить закрытие):** с помощью этого атрибута можно включить или отключить использование кнопки в строке заголовка окна для закрытия окна в среде исполнения.
- **Foreground (На переднем плане):** с помощью этого атрибута можно включить или отключить отображение окна всегда на переднем плане.

Атрибуты отображаются и настраиваются в графическом дизайнерае.

14.2.4 Редактирование макросов

Введение

Каждый из макросов в проекте отображается на своей строке в окне GSC Runtime (Среда исполнения макросов). Можно открыть макрос в окне GSC Runtime (Среда исполнения макросов), а затем отредактировать его в редакторе Global Script. После сохранения отредактированного макроса он используется в среде исполнения.

Процедура

1. Откройте контекстное меню для необходимого макроса.
2. Выберите Edit (Изменить).

Дополнительные источники информации

Создание и редактирование макросов (стр. 71)

14.3 GSC Diagnose (Диагностика макросов)

14.3.1 GSC Diagnose (Диагностика макросов)

Описание функций

GSC Diagnose (Диагностика макросов) выводит операторы printf (содержащиеся в макросах) в том порядке, в котором они вызываются в окне диагностики. Это также относится к операторам printf в функциях, которые вызываются в макросах. При тщательно продуманном использовании операторов printf, например, для вывода значений тегов, можно следить за потоком макросов и вызываемых функций. Даже состояния ошибки, которые приводят к вызову функции OnErrorExecute, отображаются в окне GSC Diagnose (Диагностика макросов).

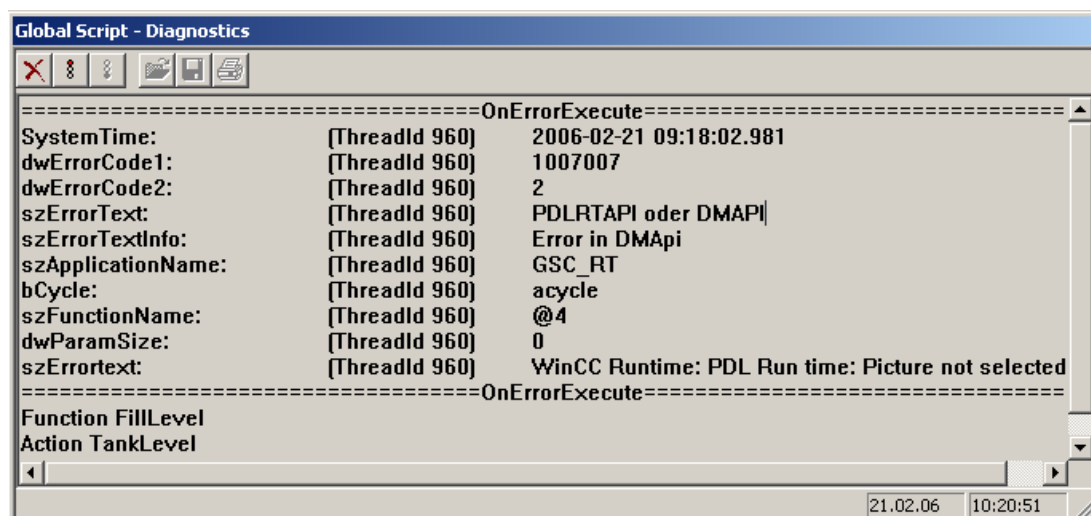


Рисунок 14-1 Окно диагностики GSC Diagnose (Диагностика макросов)

Для использования GSC Diagnose (Диагностика макросов) сначала необходимо добавить окно приложения типа GSC Diagnose (Диагностика макросов) в кадр процесса. С помощью атрибутов GSC Diagnose (Диагностика макросов) можно определить внешний вид окна GSC Diagnose (Диагностика макросов).

При изменении кадра содержимое окна GSC Diagnose (Диагностика макросов) удаляется.

Примечание

Оператор printf() может содержать до 360 символов.

Дополнительные источники информации

Панель инструментов GSC Diagnose (Диагностика макросов) (стр. 107)

Атрибуты GSC Runtime (Среда исполнения макросов) (стр. 104)

Помещение GSC Diagnose (Диагностика макросов) в кадр процесса (стр. 106)

14.3.2 Помещение GSC Diagnose (Диагностика макросов) в кадр процесса

Введение

Чтобы использовать GSC Diagnose (Диагностика макросов), необходимо добавить GSC Diagnose (Диагностика макросов) в кадр процесса. Этим кадром процесса может быть существующий кадр или кадр, служащий исключительно в целях диагностики. GSC Diagnose (Диагностика макросов) не удастся добавить в кадр процесса напрямую, ее следует добавить в качестве приложения в окне приложения. Окно приложения является частью кадра процесса. Описанные здесь действия необходимо выполнить в графическом дизайнера.

Требования

Должен быть запущен графический дизайнер и открыт кадр процесса.

Процедура

1. На палитре объектов выберите Smart Object\Application Window (Интеллектуальный объект\окно приложения).
2. В области отрисовки откройте окно приложения.
3. В диалоговом окне Window Contents (Содержимое окна) выберите Global Script.
4. Подтвердите настройки, нажав ОК.
5. В диалоговом окне Template (Шаблон) выберите GSC Diagnose (Диагностика макросов).
6. Нажмите кнопку ОК, чтобы подтвердить выбор.

Дополнительные источники информации

Атрибуты GSC Runtime (Среда исполнения макросов) (стр. 104)

Панель инструментов GSC Diagnose (Диагностика макросов) (стр. 107)

14.3.3 Атрибуты GSC Diagnose (Диагностика макросов)

Атрибуты диагностики макросов

GSC Diagnose (Диагностика макросов) имеет атрибуты, с помощью которых можно определить внешний вид окна GSC Diagnose (Диагностика макросов) в среде исполнения. Сюда входят атрибут геометрии и, в частности, следующие атрибуты:

- Display (Отображение): с помощью этого атрибута можно отобразить или скрыть окно. Атрибут можно сделать динамическим, выбрав имя Visible (Видимый).
- Sizeable (Разрешить изменение размера): с помощью этого атрибута можно разрешить или запретить изменение размера окна в среде исполнения.
- Moveable (Разрешить перемещение): с помощью этого атрибута можно разрешить или запретить перемещение окна в среде исполнения.
- Border (Граница): с помощью этого атрибута можно включить или отключить границу для окна. Если окно будет иметь границу, его ширину и высоту можно будет изменить в среде исполнения.
- Title (Заголовок): с помощью этого атрибута можно включить или отключить использование строки заголовка в окне.
- Can be Maximized (Разрешить разворачивание): с помощью этого атрибута можно включить или отключить использование кнопки в строке заголовка окна для разворачивания окна в среде исполнения.
- Can be Closed (Разрешить закрытие): с помощью этого атрибута можно включить или отключить использование кнопки в строке заголовка окна для закрытия окна в среде исполнения.
- Foreground (На переднем плане): с помощью этого атрибута можно включить или отключить отображение окна всегда на переднем плане.

Атрибуты отображаются и настраиваются в графическом дизайнера.







14.3.4 Панель инструментов GSC Diagnose (Диагностика макросов)

Функции панели инструментов

Панель инструментов GSC Diagnose (Диагностика макросов) содержит кнопки для управления выходными данными, отображающимися в окне диагностики, а также для сохранения, печати и открытия содержимого окна.



На панели инструментов содержатся кнопки для вызова следующих функций:

Кнопка	Функция
	Удаление содержимого окна.
	Остановка обновления окна.
	Возобновление обновления окна.
	Открытие текстового файла в окне.
	Сохранение содержимого окна в текстовом файле.
	Печать содержимого окна.

Индекс

A

Action

- Cause for non-execution of an action, 57
- Compiling, 57, 65
- Creating functions and actions, 7, 57
- Creating new actions, 61
- Defining, 7
- Deleting, 36
- Differences between actions and functions, 57 edit, 57
- Editing, 61, 86
- Exporting, 76
- Finding, 57
- Importing, 77
- Printing, 39
- Protection against read and write access, 64
- Renaming, 57, 77
- Runtime behavior, 81
- Saving, 65
- Structuring, 7
- Use of DLLs, 25
- Using actions from outside the project, 79

Action icon

- Features, 57

Adding, 69, 71

Adding Timer trigger, 69

Adding trigger, 71

Authorization

- Assigning, 75

Authorizations, 75

C

Color code, 30

Computer properties, 21

D

Design tool, 7

E

Edit window

- Working in the edit window, 30

Editing function, 32

- Editing functions with the keyboard, 31
- Editing functions with the mouse, 32

F

Features, 57

File, 39

- Searching, 39

Font style

- Setting, 35

Function

- Application, 43
- Compiling, 38, 43, 52
- Creating, 7, 43
- Creating new function, 45
- Defining, 7
- Differences between actions and functions, 57
- Editing, 43
- Finding, 43
- Functions from other sources, 54
- Printing, 39
- Protection against read and write access, 51
- Renaming, 53
- Saving, 52
- Structuring, 7
- Use of DLLs, 25
- Using internal functions, 48

Function code, 47

- Writing, 47

G

Global action, 19

- Application, 19
- Characteristics, 19

Global C-tags, 23

- Application, 23
- Definition, 23

- Global Script, 27
 - Action icon, 57
 - Action structure, 7
 - Adding action-related information, 62
 - Adding function-related Information, 50
 - Assigning authorization, 75
 - Authorizations, 75
 - Cause for non-execution of an action, 57
 - Changing trigger, 72
 - Characteristics of global actions, 19
 - Characteristics of internal functions, 15
 - Characteristics of local actions, 17
 - Characteristics of project functions, 11
 - Characteristics of standard functions, 13
 - Color code, 30
 - Compiling action, 57, 65
 - Compiling function, 38, 43, 52
 - Computer properties, 21
 - Creating action, 7, 57
 - Creating function, 7
 - Creating internal function, 45
 - Creating new actions, 61
 - Creating new function, 45
 - Defining action, 7
 - Defining function, 7
 - Definition of global C-tags, 23
 - Deleting action, 36
 - Deleting project functions, 36
 - Deleting standard functions, 36
 - Deleting trigger, 74
 - Design tool, 7
 - Differences between actions and functions, 57
 - Editing action, 57, 61, 86
 - Editing function, 43
 - Editing functions with the keyboard, 32
 - Editing functions with the mouse, 32
 - Effect of triggers on actions, 66
 - Exporting action, 76
 - Features, 57
 - Finding actions, 57
 - Finding functions, 43
 - Function structure, 7
 - Generating new header, 37
 - GSC Diagnose toolbar, 89
 - GSC Runtime, 82
 - GSC Runtime attributes, 86
 - Identification of actions in the navigation window, 66
 - Import action, 77
 - Including Global Script Runtime in the startup list of the project, 21
 - Including GSC Runtime in a process picture, 85
 - Inserting GSC Diagnose in a process picture, 88
 - New trigger of type Trigger:new trigger of type Global Script:Adding trigger, 69
 - New trigger of type Trigger:New trigger of type TagGlobalScript:Adding trigger, 71
 - Opening, 77
 - Opening page view, 41
 - Password input, 51, 64
 - Printing action, 39
 - Printing function, 39
 - Printing project documentation, 41
 - Properties, 50, 51, 62, 64, 72, 74
 - Protecting function against read and write access, 51
 - Protection against read and write access, 64
 - Renaming action, 57, 77
 - Renaming function, 53
 - Runtime behavior of actions, 81
 - Saving action, 65
 - Saving file as, 76
 - Saving function, 52
 - Searching files, 39
 - Setting different views, 35
 - Setting print parameters, 40
 - Setting the font style, 35
 - Structure of Global Script editor, 27
 - System behavior, 57
 - Trigger type, 7
 - Use of DLLs in actions, 25
 - Use of DLLs in functions, 25
 - Use of global actions, 19
 - Use of global C-tags, 23
 - Use of Internal Functions, 15
 - Use of local actions, 17
 - Use of project functions, 11
 - Use of standard functions, 13
 - Using "Save As...", 36
 - Using actions from other sources, 79
 - Using functions, 43
 - Using Functions from other sources, 54
 - Using internal functions, 48
 - Using project functions, 49
 - Using standard functions, 49
 - Validity range, 23
 - WinCC coding rules, 60
 - Working in the edit window, 30
 - Working with toolbars, 32
 - Writing function code, 47
- Global Script editor, 27
 - Structure, 27
- Global Script Runtime, 21
 - Adding to startup list of the project, 21
- GSC Diagnose

Including in process picture, 88
Toolbar, 89
GSC Runtime
Attributes, 86
Including in process picture, 85

H

Header, 37
Regenerating, 37

I

Information
Adding action-related information, 62
Internal function, 15
Application, 15
Characteristics, 15

L

Local actions, 17
Application, 17
Characteristics, 17

M

Menu bar, 27

N

Navigation window
Identification of actions in the navigation window, 66

O

Opening, 77
Output window, 27

P

Page view
Opening, 41
Password input, 51, 64

Print parameters, 40
Setting, 40
Project documentation
Printing, 41
Project function, 11
Application, 11
Applying, 49
Characteristics, 11
Deleting, 36
Properties, 50, 51, 62, 64, 72, 74

S

Save As...
Applying, 36
Saving file as, 76
Standard function
Application, 13
Applying, 49
Characteristics, 13
Deleting, 36
Status bar, 27
System behavior, 57

T

Toolbar
Working with toolbars, 32
Trigger
Changing, 72
Deleting, 74
Effect on actions, 66
Trigger type, 7, 66

V

Validity range, 23
View, 35
Setting different views, 35

W

WinCC coding rules, 60