



*Allen-Bradley*

**Руководство для  
ускоренной  
подготовки к работе  
с семейством  
программируемых  
контроллеров  
Logix5000™**

1756 ControlLogix®  
1769 CompactLogix  
1789 SoftLogix  
1794 FlexLogix , PowerFlex  
700S with DriveLogix

**Упрощенное руководство**

**Rockwell  
Automation**

## Важная информация для пользователя

Эксплуатационные характеристики полупроводникового оборудования отличаются от характеристик электромеханического оборудования. В публикации SGI\_1.1 фирмы Allen\_Bradley "Руководство по обеспечению безопасности при использовании, установке и обслуживании полупроводниковых устройств управления", имеющейся в вашем местном представительстве Rockwell Automation, а также в Интернете по адресу <http://www.ab.com/manuals/gi>, описываются некоторые важные различия между полупроводниковым оборудованием и электромеханическими устройствами с жесткими соединениями. В связи с этими различиями, а также с большим разнообразием применений полупроводникового оборудования, все лица, ответственные за использование такого оборудования, должны быть уверены в его применимости.

Rockwell Automation ни в коем случае не отвечает за косвенный ущерб, связанный с использованием такого оборудования.

Примеры и схемы приводятся в данном руководстве исключительно для иллюстрации. Поскольку каждое конкретное оборудование характеризуется множеством специфических параметров и требований, Rockwell Automation, Inc. не берет на себя ответственность за фактическое использование продуктов на основе таких примеров и схем.

Rockwell Automation, Inc. не несет патентную ответственность в связи с использованием информации, цепей, оборудования или программного обеспечения, описанных в данном руководстве.

Воспроизведение содержания данного руководства, целиком или частично, без письменного разрешения Rockwell Automation, Inc. запрещается.

В настоящем документе используются примечания, обращающие ваше внимание на вопросы безопасности.

### ПРЕДУПРЕЖДЕНИЕ



Обозначает информацию о способах действий или обстоятельствах, которые могут привести к взрыву в опасных условиях, что может повлечь травмы или смерть людей, материальный ущерб или экономические потери.

### ВНИМАНИЕ



Обозначает информацию о способах действий или обстоятельствах, которые могут привести к травмам или смерти людей, материальному ущербу или экономическим потерям. Такие примечания помогут вам:

- обнаружить опасность
- избежать опасности
- понять последствия

### ВАЖНО

Обозначает информацию, имеющую критическое значение для успешного применения и понимания продукта.

### ОПАСНОСТЬ ПОРАЖЕНИЯ ТОКОМ



Такие знаки могут быть нанесены снаружи или внутри устройства для предупреждения о возможном наличии опасного напряжения.

**Введение**

В данной версии документа содержится новая и обновленная информация. Такая информация обозначается вертикальной полосой, как показано слева от данного абзаца.

**Новая или обновленная информация**


Данный документ содержит следующие изменения:

<b>Данное изменение:</b>	<b>Начинается на стр.:</b>
Программирование и тестирование простого проекта - новая глава, которая описывает минимальные требования по программированию контроллера S <sup>+</sup> * 5000	1-1
Экспорт/импорт релейной логики	3-6
Ввод последовательной функциональной схемы	3-16
Присваивание операндов инструкции для структурированного текста	3-15
Описание типа данных, задаваемого пользователем	4-2
Добавление комментариев к цепочке	4-4
Ввод комментариев к цепочке при помощи Microsoft® Excel	4-5
Добавление комментариев к функциональной блок-схеме или ПФС	4-7
Добавление комментариев к структурированному тексту	4-9
Редактирование логики в режиме онлайн – добавление ПФС или структурированного текста	6-1
Подтверждение всех редакторских правок в программе	6-5
Просмотр логики на предмет поиска тега, инструкции, комментария и т.д.	7-7

**Для заметок:**

**Когда  
пользоваться  
данным  
руководством**

Данное руководство является одним из множества документов, предназначенных для работы с Logix5000

Вы здесь 

<b>Чтобы:</b>	<b>Используйте:</b>
Начать работать с контроллером Logix5000	Руководство для ускоренной подготовке к работе с <i>Logix5000™</i> , публикация 1756-QS001
Найти аббревиатуры и методики относящиеся к языкам программирования, инструкциям, соединениям и состояниям	<i>Системный справочник для контроллеров Logix5000</i> , публикация 1756-QR007
Программировать контроллер Logix5000 с учетом детальной и исчерпывающей информации	Общая методика для программируемых контроллеров <i>Logix5000 Controllers Common Procedures</i> , публикация 1756-PM001
Программировать отдельные инструкции Logix5000	<ul style="list-style-type: none"> <li>• Справочное руководство по общим инструкциям для контроллеров <i>Logix5000</i>, публикация 1756-RM006</li> <li>• Справочник инструкций по управлению технологическим процессом и приводами для контроллеров <i>Logix5000</i>, публикация 1756-RM006</li> <li>• Справочник инструкций по выполнению перемещений для контроллеров <i>Logix5000</i>, публикация 1756-RM007</li> </ul>
Импортировать или экспортировать проект Logix5000 или теги из текстового файла	<i>Справочное руководство по экспорту/импорту для контроллеров Logix5000</i> , публикация 1756-RM084
Преобразовать приложения PLC-5 или SLC 500 в проект Logix5000	<i>Справочное руководство по преобразованию алгоритмов приложений PLC-5 или SLC 500 для контроллера Logix5550</i> , публикация 1756-6.8.5
Интегрировать заданный контроллер Logix5000 в систему контроллеров, модулей ввода/вывода и других устройств	<ul style="list-style-type: none"> <li>• Руководство пользователя для <i>CompactLogix System</i>, публикация 1769-UM007</li> <li>• Руководство пользователя для <i>ControlLogix System</i>, публикация 1756-UM001</li> <li>• Руководство пользователя для контроллера <i>DriveLogix</i>, публикация 20D-UM002</li> <li>• Руководство пользователя для <i>FlexLogix System</i>, публикация 1794-UM001</li> <li>• Руководство пользователя для <i>SoftLogix5800 System</i>, публикация 1789-UM002</li> </ul>
Управлять устройствами при помощи сети EtherNet/IP	<i>Модули EtherNet/IP для систем управления Logix5000, руководство пользователя</i> , публикация ENET-UM001
Управлять устройствами при помощи сети ControlNet™	<i>Модули ControlNet для систем управления Logix5000, руководство пользователя</i> , публикация CNET-UM001
Управлять устройствами при помощи сети DeviceNet™	<i>Модули DeviceNet для систем управления Logix5000, руководство пользователя</i> , публикация DNET-UM004

## Назначение данного руководства

Данное руководство представляет набор начальных процедур для:

- установки связи с контроллером Logix5000
- программирования контроллера Logix5000
- выполнения вспомогательных задач в режиме онлайн, таких как поиск и редактирование алгоритма, запуск гистограммы, стирание ошибок и форсировка значений ввода/вывода.

Термин «*контроллер Logix5000*» относится к контроллерам:

- 1756 ControlLogix®
- 1769 CompactLogix™
- 1789 SoftLogix5800™
- 1794 FlexLogix™
- PoweFlex®700S с контроллерами DriveLogix™

## Для кого предназначено это руководство

Данное руководство предназначено для лиц, занимающихся программированием или обслуживанием систем управления производством.

Чтобы пользоваться данным руководством вы должны быть знакомы с:

- Программируемыми контроллерами
- Системами управления производством
- Персональными компьютерами и операционными системами Windows® 95, Windows 98, Windows NT® или Windows 2000

## Как пользоваться данным руководством

При использовании этого руководства вам встретятся некоторые термины, выделенные из остального текста:

Текст:	Определяет:	Например:	Означает:
<i>Выделенный курсивом</i>	название элемента, который вы видите на экране или в примере	Щелкните правой кнопкой мыши на <i>User-Defined...</i>	Следует щелкнуть правой кнопкой мыши на элементе, который называется User-Defined.
Выделенный шрифтом <code>courier</code>	информацию (параметр), которую вы должны ввести для своего приложения	Щелкните правой кнопкой мыши на <code>name_of_program...</code>	Вы должны указать конкретную программу в вашем приложении. Как правило, это заданное вами имя или параметр.
В квадратных скобках	клавиша на клавиатуре	Нажмите [Enter].	Нажмите на клавишу Enter.

<b>Программирование и тестирование простого проекта</b>	<b>Глава 1</b>	
	Использование этой главы	1-1
	Создание проекта для контроллера	1-2
	Создание проекта	1-3
	Соглашения об именах	1-3
	Добавление модулей ввода/вывода	1-4
	Просмотр данных ввода/вывода	1-5
	Ввод релейной логики	1-7
	Открытие процедуры	1-7
	Ввод релейной логики	1-8
	Ввод функциональной блок-схемы	1-9
	Создание процедуры	1-9
	Вызов процедуры	1-10
	Ввод функциональной блок-схемы	1-11
	Конфигурирование инструкции на языке функциональных блок-схем	1-12
	Присваивание тегов-псевдонимов устройствам	1-13
	Как показать или скрыть информацию о тегах-псевдонимах	1-14
	Настройка связи с контроллером через последовательный порт	1-15
	Загрузка проекта в контроллер	1-17
Выбор рабочего режима контроллера	1-19	
<b>Организация проекта</b>	<b>Глава 2</b>	
	Использование этой главы	2-1
	Конфигурирование выполнения задачи	2-2
	Создание дополнительных программ	2-3
	Создание пользовательского типа данных	2-5
	Задание процедур	2-7
	Задание процедуры для каждого участка вашей установки или процесса	2-7
	Определение установленных языков программирования	2-8
	Выбор языка программирования для каждой процедуры	2-8
	Разбивка каждой процедуры на более осмысленные части	2-9
	Определение главных процедур	2-10
	Конфигурирование контроллера	2-11
Конфигурирование ввода/вывода	2-12	
<b>Программирование проекта в режиме оффлайн</b>	<b>Глава 3</b>	
	Использование этой главы	3-1
	Ввод релейной логики	3-2
	Буксировка элемента	3-3
	Использование клавиатуры для добавления элемента	3-4
	Ввод логики с использованием текста ASCII	3-4
	Использование клавиш быстрого ввода	3-5
	Экспорт/импорт релейной логики	3-6
	Когда вы импортируете цепочки	3-6
	Экспорт цепочек	3-7
	Импорт цепочек	3-8
	Проверка тегов-псевдонимов	3-8
	Ввод функциональной блок-схемы	3-9
Использование клавиатуры для добавления элемента	3-10	

Связь элементов .....	3-10
Создание цикла .....	3-11
Добавление листа .....	3-11
Использование разметочной панели для функционального блока .....	3-12
Создание раздела .....	3-13
Добавление разметочной панели к Microsoft Excel .....	3-13
Ввод структурированного текста .....	3-14
Просмотр инструкции .....	3-15
Присваивание операндов инструкции .....	3-15
Ввод последовательной функциональной схемы .....	3-16
Ввод ПФС .....	3-17
Присваивание операндов .....	3-18
Создание тега .....	3-19
Выбор существующего тега .....	3-19
Проверка проекта .....	3-20
Руководящие указания по созданию тегов .....	3-22

#### Глава 4

### Документирование проекта

Использование данной главы .....	4-1
Описание пользовательского типа данных .....	4-2
Включение и выключение сквозных (pass-through) и присоединяемых (append) описаний .....	4-3
Вставка сквозных описаний в RSLogix .....	4-3
Добавление комментария к цепочке .....	4-4
Ввод и редактирование комментария с использованием Microsoft® Excel .....	4-5
Экспорт существующего комментирия .....	4-5
Редактирование экспортируемого файла .....	4-6
Импортирование новых комментариев .....	4-6
Добавление комментария к функциональной блок-схеме или ПФС .....	4-7
Настройка опции Word Wrap (Укладка текста) .....	4-7
Добавление текстового окна .....	4-8
Добавление комментария к структурированному тексту .....	4-9

#### Глава 5

### Работа с контроллером в режиме онлайн

Использование данной главы .....	5-1
Настройка связи с контроллером через EtherNet/IP .....	5-2
Нужное вам оборудование и информация .....	5-3
Подключение устройства EtherNet/IP к компьютеру .....	5-3
Присваивание адреса IP контроллеру или коммуникационному модулю .....	5-4
Конфигурирование драйвера Ethernet .....	5-5
Работа с контроллером в режиме онлайн .....	5-6
Если на вашем компьютере есть проект, выполняемый на контроллере... ..	5-6
Если на вашем компьютере нет проекта, выполняемого на контроллере... ..	5-7



---

<b>Программирование проекта в режиме онлайн</b>	<b>Глава 6</b>	
	Использование этой главы .....	6-1
	Редактирование логики в режиме онлайн.....	6-1
	Запуск отложенного редактирования.....	6-3
	Редактирование и принятие правок.....	6-3
	Тестирование правок.....	6-4
	Трансляция и сохранение правок .....	6-4
	Окончательное принятие всех правок в программе .....	6-5
<b>Устранение неполадок в контроллере</b>	<b>Глава 7</b>	
	Использование этой главы .....	7-1
	Диагностирование связи с устройствами ввода/вывода.....	7-2
	Сброс основной ошибки .....	7-4
	Поиск в проекте.....	7-5
	Просмотр релейной логики.....	7-7
	Форсировка значения ввода/вывода .....	7-8
Создание и запуск тренда (гистограммы) .....	7-11	
Просмотр времени сканирования.....	7-13	



# Программирование и тестирование простого проекта

## Использование этой главы

Эта глава содержит основные программные последовательности для контроллера Logix5000™.

- Она содержит шаги, которые необходимы для разработки и тестирования релейной логики или функциональной блок-схемы.
- Примеры, представленные в данной главе, показывают, как управлять цифровым или аналоговым выходом на основе состояния цифрового или аналогового входа.

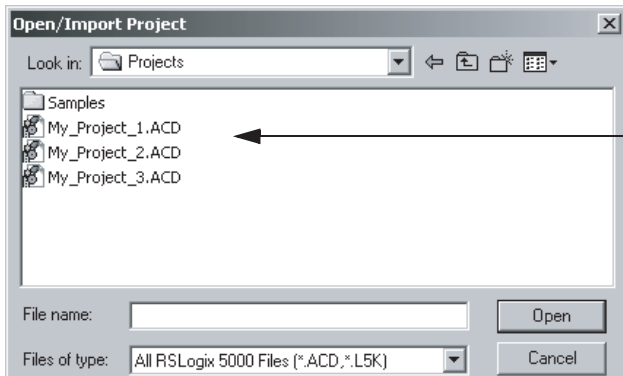
Чтобы запрограммировать и протестировать простой проект:

Шаг:	Стр.:
Создайте проект для данного контроллера	1-2
Добавьте модули ввода/вывода	1-4
Просмотрите данные ввода/вывода	1-5
Введите релейную логику	1-7
Введите функциональную блок-схему	1-9
Присвойте псевдонимы тегов для ваших устройств	1-13
Установите последовательное соединение с контроллером	1-15
Загрузите проект в контроллер	1-17
Выберите режим работы контроллера	1-18

Другие главы данной публикации содержат более подробную информацию о том, как запрограммировать, отредактировать и тестировать проект на наличие ошибок.

## Создание проекта для контроллера

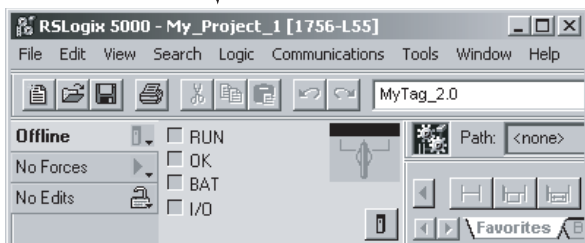
Чтобы сконфигурировать и запрограммировать контроллер Logix5000 вам необходимо использовать программное обеспечение RSLogix™ 5000.



**Проект** - файл на вашей рабочей станции (или сервере) в котором хранится релейная логика, конфигурация, данные и документация для контроллера.

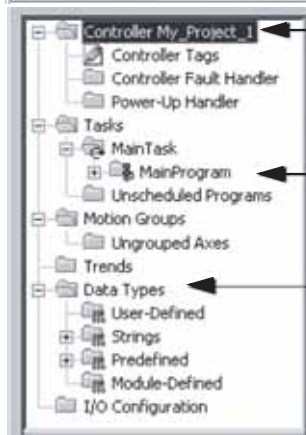
- Файл проекта имеет расширение .ACD.
- При создании проекта имя проекта совпадает с именем соответствующего контроллера.
- Имя контроллера не зависит от имени проекта. Вы можете переименовать либо проект, либо контроллер.

Имя проекта



Если вы переименуете проект или контроллер, будут показаны оба имени.

Имя контроллера



**Организатор контроллера** - графическое представление проекта. Используйте организатор проекта для перемещения по различным элементам проекта.

Чтобы открыть папку и просмотреть содержание:

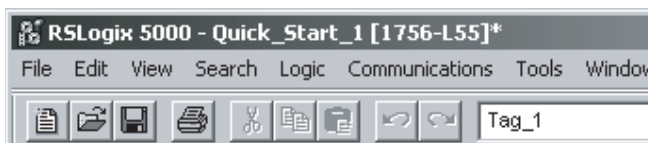
- Дважды щелкните на нужной папке
- Щелкните на символе "+".

Чтобы закрыть папку и спрятать ее содержание:

- Дважды щелкните на нужной папке
- Щелкните на символе "-".

## Создание проекта

1. Запустите программу  
RSLogix 5000



2. Щелкните на кнопке New (Новый)

3. Задайте общую конфигурацию контроллера (некоторые пункты применимы только для определенных контроллеров)

**New Controller**

Vendor: Allen-Bradley

Type: 1756-L55 ControlLogix5555 Controller

Revision:

Redundancy Enabled

Name:

Description:

Chassis Type: 1756-A10 10-Slot ControlLogix Chassis

Slot:

Create In: C:\RSLogix 5000\Projects

- a. Тип контроллера

- b. Основную версию микропрограммного обеспечения для данного контроллера

- c. Имя контроллера

- d. Размер шасси контроллера

- e. Номер слота контроллера

- f. Папку для сохранения проекта

4. Нажмите

## Соглашения об именах:

В рамках проекта Logix5000 вы задаете имена для различных его элементов, таких как контроллер, адреса данных (теги), пути, модули ввода/вывода и т.д. При вводе имен придерживайтесь следующих правил:

- Используйте только буквы, цифры и знаки подчеркивания ( \_ )
- Имена должны начинаться с буквы или знака подчеркивания
- Должны содержать не более 40 знаков
- Имена не чувствительны к регистру

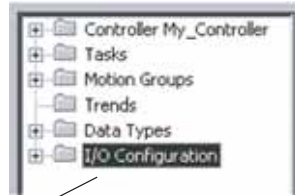
## Добавление модулей ввода/вывода

Для организации связи с модулями ввода/вывода вам необходимо добавить эти модули в папку I/O Configuration данного контроллера. Свойства, которые вы выберете для каждого модуля, определяют работу данного модуля.

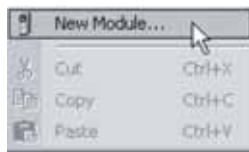
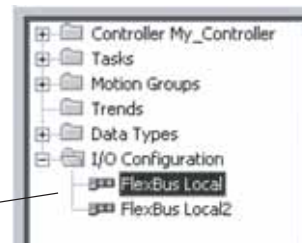
Контроллер CompactLogix



Контроллер ControlLogix

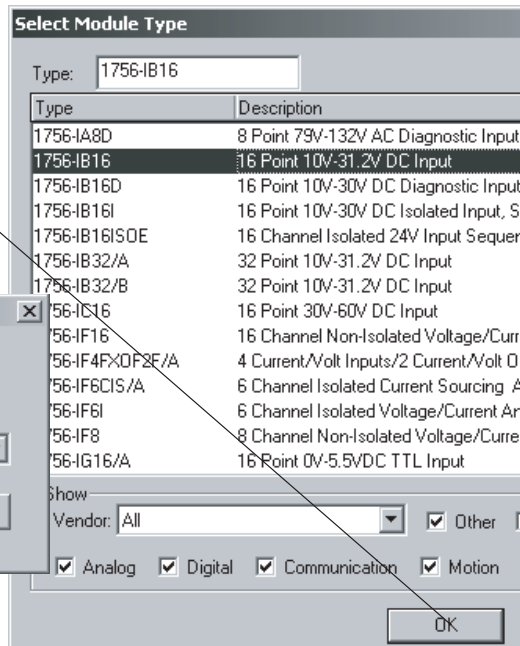


Контроллер FlexLogix

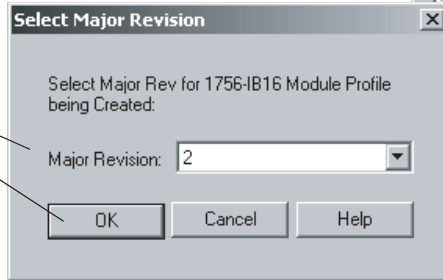


1. Щелкните правой клавишей мыши и выберите *New Module* (Новый модуль).

2. Выберите тип модуля.



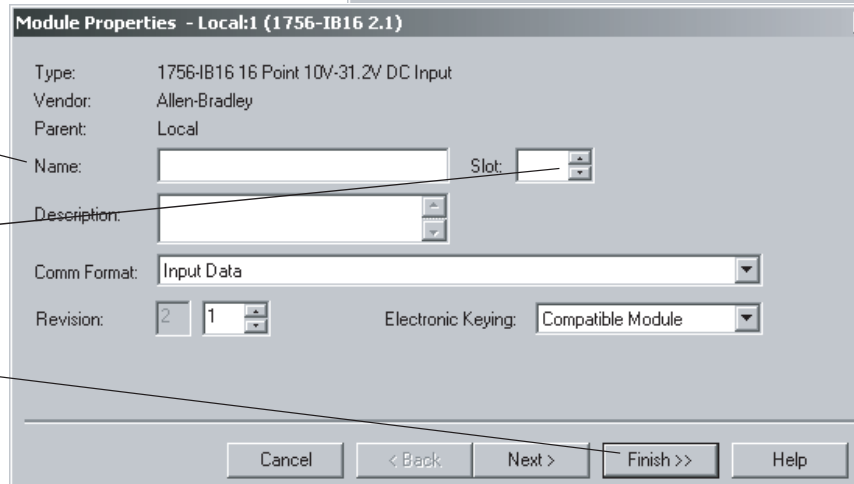
3. Выберите версию модуля.



4. Введите с клавиатуры имя для модуля (до 40 символов без пробелов).

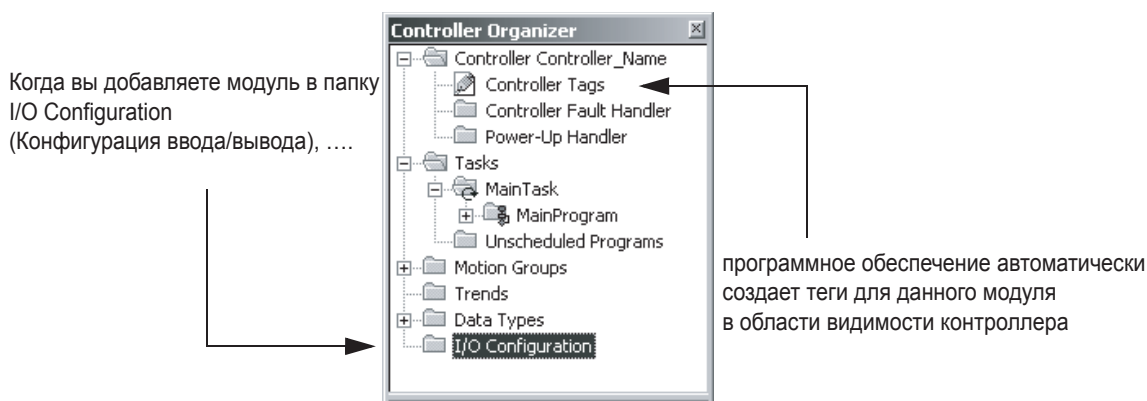
5. Выберите положение модуля на шасси или шине.

6. Согласитесь с конфигурацией "по умолчанию" для данного модуля.



## Просмотр данных ввода/вывода

Информация ввода/вывода представляется в виде набора тегов.

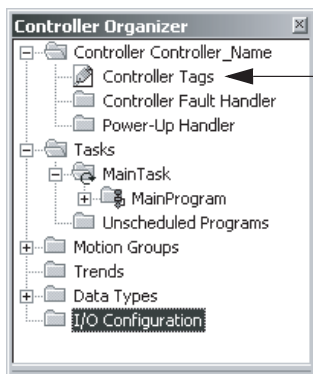


Адрес ввода/вывода имеет следующий формат:

*Location* *:Slot* *:Type* *.Member* *.SubMember* *.Bit*

= Необязательно

Где:	Это:
<i>Location</i>	Местоположение в сети LOCAL = то же шасси или рейка DIN, что и у контроллера ADAPTER_NAME = обозначает удаленный коммуникационный адаптер или модуль моста
<i>Slot</i>	Номер слота модуля ввода/вывода в соответствующем шасси или рейке DIN
<i>Type</i>	Тип данных I = ввод O = вывод C = конфигурация S = состояние
<i>Member</i>	Конкретные данные, поступающие от модуля ввода/вывода; зависят от того, данные какого типа могут храниться в данном модуле. <ul style="list-style-type: none"> <li>Для цифрового модуля, в члене Data (Данные) обычно хранятся двоичные значения ввода или вывода.</li> <li>Для аналогового модуля, в члене Channel (CH#) (Канал) обычно хранятся данные для канала.</li> </ul>
<i>SubMember</i>	Конкретные данные, относящиеся к члену (Member).
<i>Bit</i>	Конкретная точка в модуле цифрового ввода/вывода; зависит от размера модуля ввода/вывода (0-31 для 32-точечного модуля).



1. Щелкните правой клавишей мыши и выберите *Monitor Tags (контроль тегов)*.

Значения будут показаны в следующем виде:

Вид	Основание	Обозначение
двоичный	2	2#
десятичный	10	NA
шестнадцатеричный	16	16#
восьмеричный	8	8#
экспоненциальный	NA	0.0000000e+000
с плавающей точкой	NA	0.0

Синяя стрелка показывает, что когда вы измените значение, это немедленно вступит в силу.

Tag Name	Value	Force Mask	Style
+ Local:0:C	{...}	{...}	
+ Local:0:I	{...}	{...}	
- Local:0:O	{...}	{...}	
- Local:0:O.Data	2#000...		Binary
- Local:0:O.Data.0	0		Decimal
- Local:0:O.Data.1	0		Decimal
- Local:0:O.Data.2	0		Decimal

2. Чтобы увидеть значение в другом виде, выберите желаемый для вас вид.

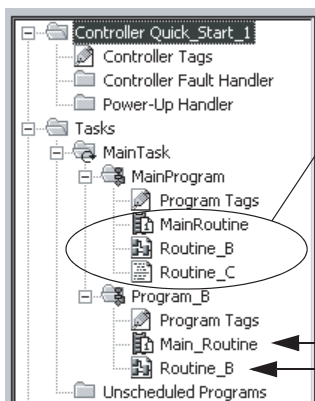
3. Чтобы изменить значение щелкните на ячейке Value (Значение) и нажмите клавишу [Enter].

4. Чтобы развернуть тег и просмотреть его элементы, щелкните на значке +.



## Ввод релейной логики

Для контроллера Logix5000 алгоритм вводится в виде процедур.



**процедура (routine)** - обеспечивает исполняемый код (алгоритм) для программы (подобно программному файлу в контроллере PLC или SLC).

**главная процедура (main routine)** - каждой программе вы присписываете главную процедуру.

- Когда программа выполняется, автоматически выполняется ее главная процедура.
- Используйте главную процедуру для управления выполнением других процедур в подпрограмме.
- Для вызова (выполнения) другой процедуры (подпрограммы) из программы, используйте инструкцию Jump to Subroutine (JSR) (переход к подпрограмме).

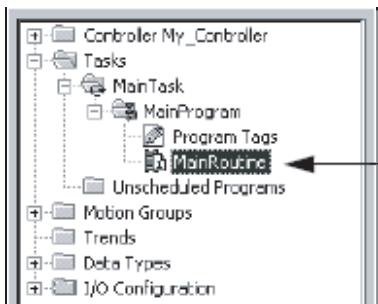
**подпрограмма (subroutine)** - всякая процедура, не являющаяся главной процедурой и процедурой обработки ошибок. Для выполнения подпрограммы используйте инструкцию Jump to Subroutine (JSR) в другой процедуре, например, в главной процедуре.

## Открытие процедуры

Когда вы создаете проект, программное обеспечение автоматически создает главную процедуру, которая использует язык релейных схем.

Чтобы открыть папку и показать ее содержимое, выполните одно из двух действий:

- Дважды щелкните по соответствующей папке.
- Щелкните на значке +.



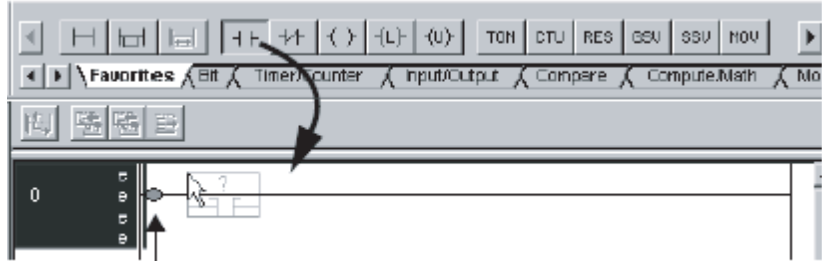
Чтобы открыть процедуру, дважды щелкните по соответствующей процедуре.

### Ввод релейной логики

Одним из способов ввода релейной логики является буксировка кнопок из панели инструментов в нужное место.

Чтобы ввести релейную логику, отбуксируйте необходимую для цепочки кнопку или инструкцию прямо в нужное место.

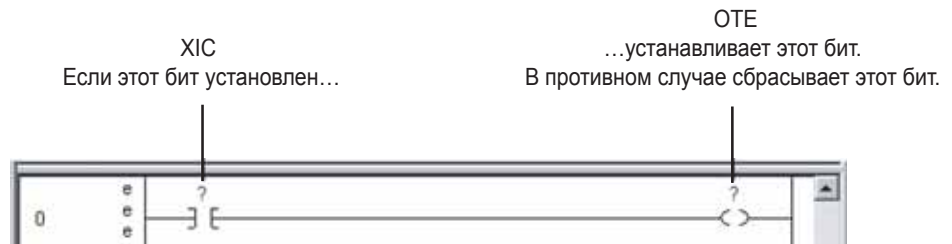
Вы можете ввести ваш алгоритм и оставить операнды не определенными. После того, как вы введете часть алгоритма, вернитесь назад и присвойте операнды.



Зеленая точка указывает допустимое место для размещения (точку опускания).

### ПРИМЕР

В данном примере инструкция Examine If Closed (XIC) (проверка на замыкание) проверяет состояние пусковой кнопки (включено/выключено). Если кнопка включена (замкнута) инструкция Output Energize (OTE) включает свет.



## Ввод функциональной блок-схемы

## Создание процедуры

В каждой процедуре вашего проекта используется определенный язык программирования. Для программирования на другом языке необходимо создать новую процедуру.

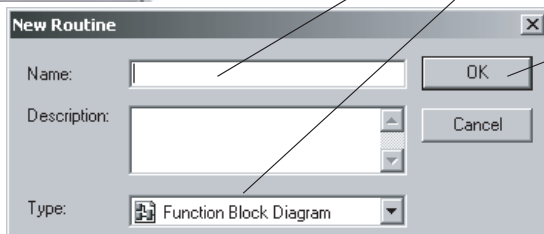


1. Щелкните правой клавишей мыши, затем на программе и выберите *New Routine (Новая Процедура)*.

2. Введите с клавиатуры имя этой процедуры.

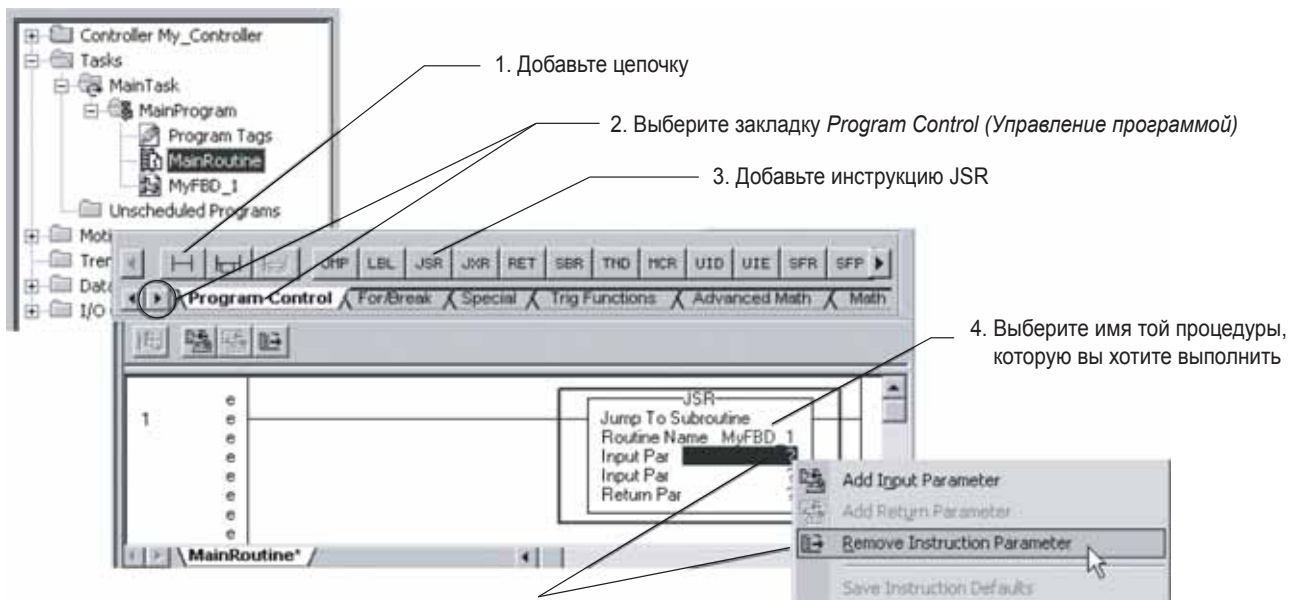
3. Выберите язык программирования.

4. OK



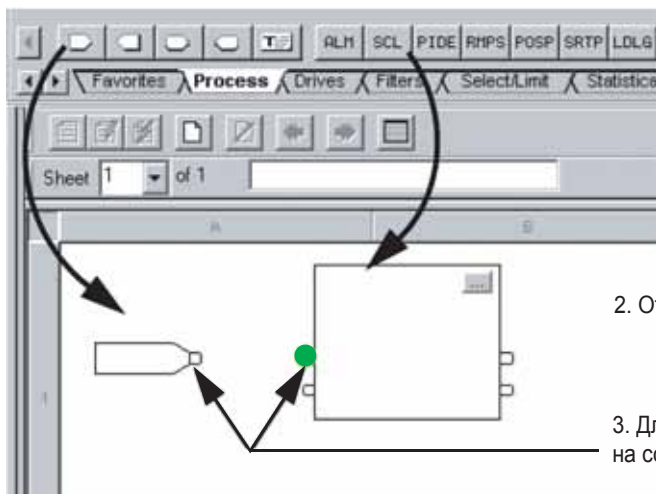
## Вызов процедуры

В случае, если процедура не является главной, для вызова процедуры используется инструкция Jump to Subroutine (JSR) (переход к процедуре).



5. Для того, чтобы просто ввести процедуру, удалите оставшиеся параметры для данной инструкции JSR. Для удаления параметров щелкните правой клавишей мыши на данном параметре, и выберите *Remove Instruction Parameter* (*Удалить параметр инструкции*).

### Ввод функциональной блок-схемы



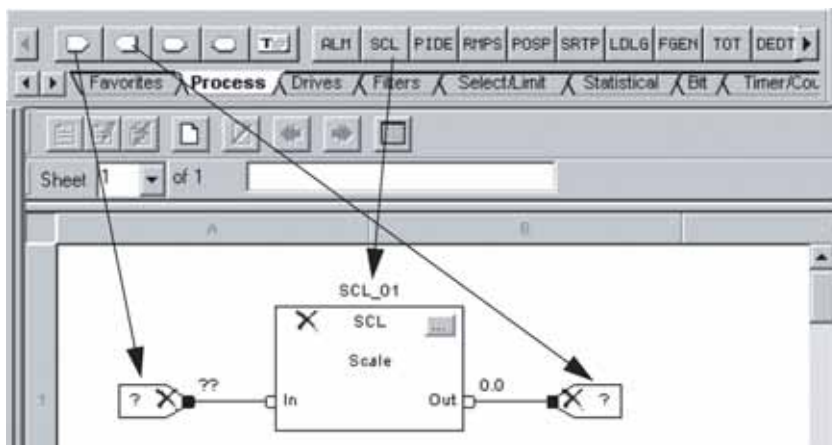
1. Щелкните на закладке нужной инструкции

2. Отбуксируйте элемент из панели инструментов на панель

3. Для подключения элементов щелкните на соответствующих контактах (зеленая точка = точка связи)

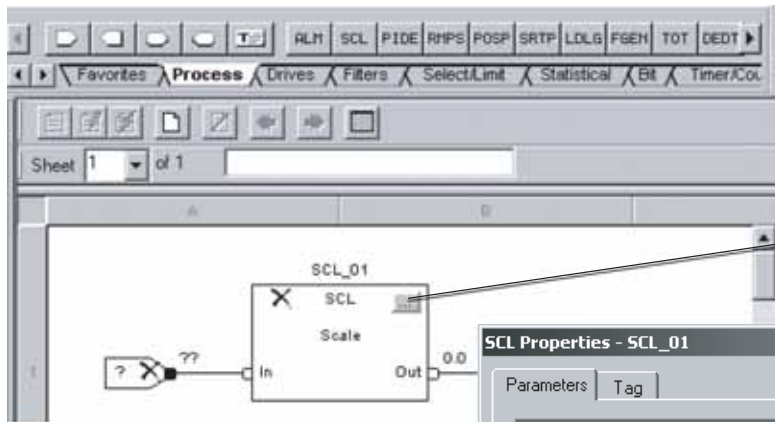
#### ПРИМЕР

В предложенном ниже примере Input Reference (IREF) считывает значение аналогового входа и посылает это значение инструкции Scale (SCL). Инструкция SCL преобразует данное значение в инженерный формат и посылает в Output Reference (OREF). OREF записывает данное значение в аналоговый выход.



## Конфигурирование инструкции на языке функциональных блок-схем

Для того, чтобы присвоить заданные значения (параметры) функциональному блоку:



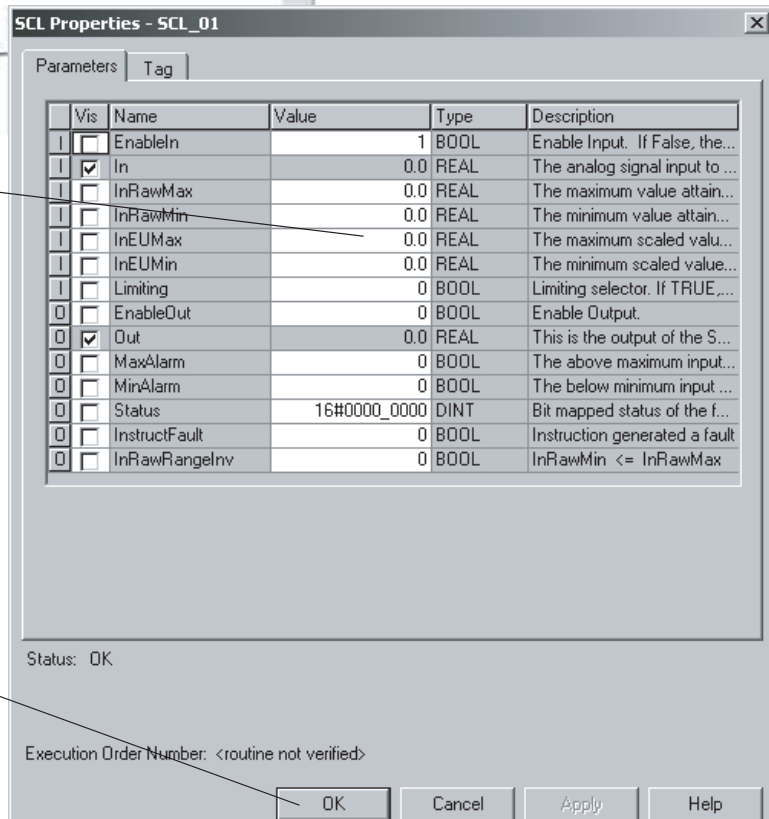
1. Щелкните на кнопке конфигурации.

2. Чтобы изменить значение параметра щелкните на ячейке данного параметра, введите новое значение с клавиатуры и нажмите [Enter].

Например, в инструкции SCL задайте следующие параметры:

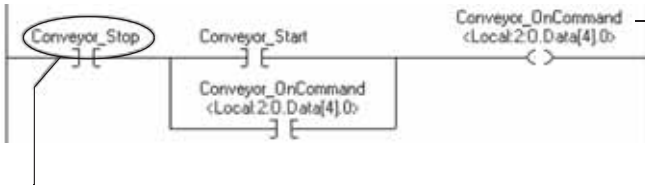
- InRawMax - максимальное значение входной величины
- InRawMin - минимальное значение входной величины
- InEUMax - максимальное значение в инженерном формате
- InEUMin - значение в инженерном формате значение в инженерном

3. OK



## Присваивание тегов-псевдонимов устройствам

Хотя вы и можете использовать входные и выходные теги модуля прямо в алгоритме, удобнее использовать псевдонимы тегов.



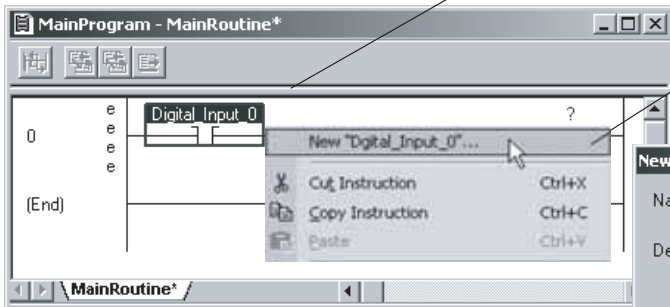
В качестве необязательной функции, создайте теги, которые описывают каждое устройство без указания в них на конкретные адреса этих устройств. Позднее, преобразуйте эти теги в псевдонимы для данных от этих устройств.

**Тег-псевдоним** - это тег, который представляет другой тег

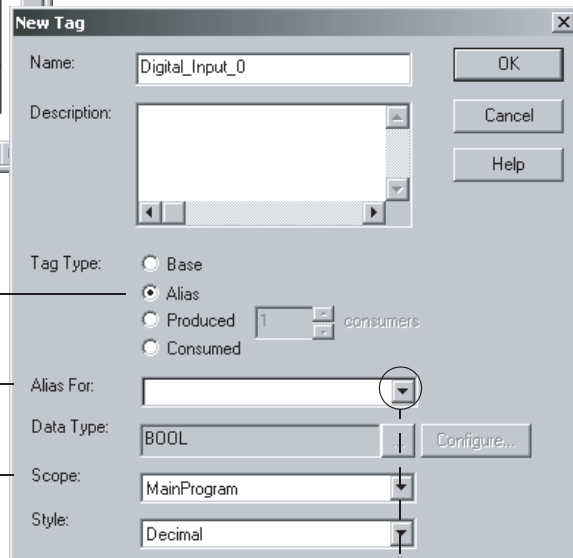
- Оба тега имеют они и те же данные.
- Когда эти данные изменяются, изменяются оба тега.
- Тег-псевдоним обеспечивает описательное имя для данных, таких как вход DeviceNet или выходные данные.
- Если положение данных меняется, просто укажите тегу на новое положение без редактирования алгоритма.

1. Введите вашу логику

2. Введите с клавиатуры описательное имя для устройства.



3. Щелкните правой клавишей мыши на имени тега и выберите New... (новый)



4. Выберите кнопку *Alias* (псевдоним).

5. Выберите тег, который представляет данный тег-псевдоним.

6. Выберите область видимости для этого тега-псевдонима.

7. Выберите OK

Просмотрите теги в видимости контроллера.



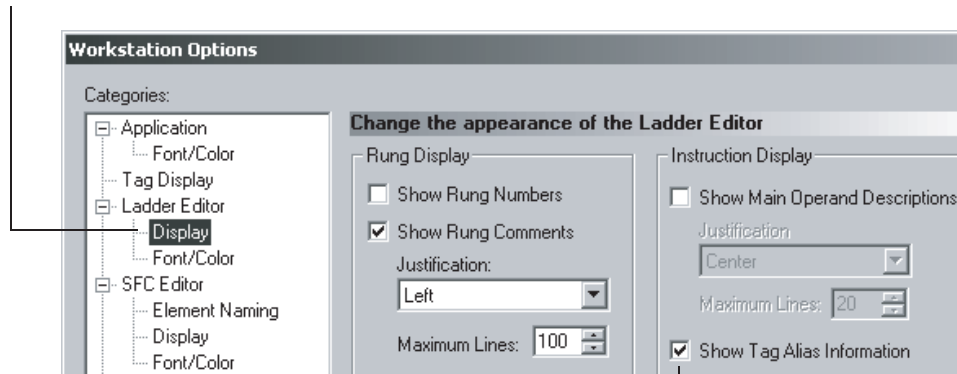
Выберите адрес данных. Чтобы выбрать разряд, щелкните ▼.

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

## Как показать или скрыть информацию о тегах-псевдонимах

Для того, чтобы показать или спрятать информацию о теге:

1. Выберите *Tools (Инструментарий) => Options (Опции)*.
2. Выберите категорию *Ladder Editor Display*.



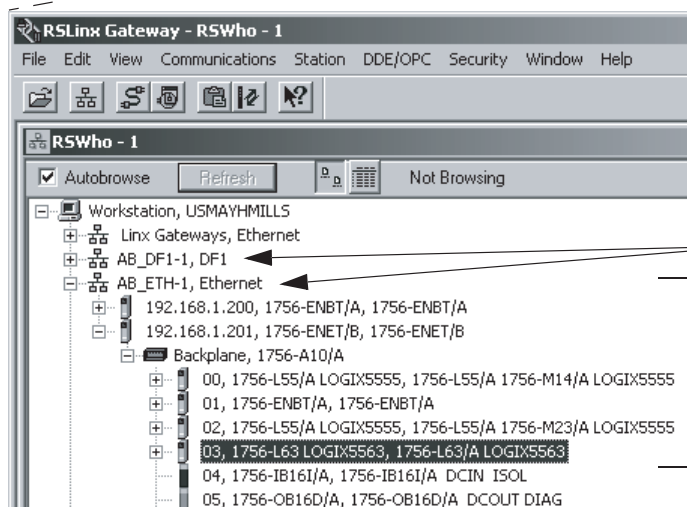
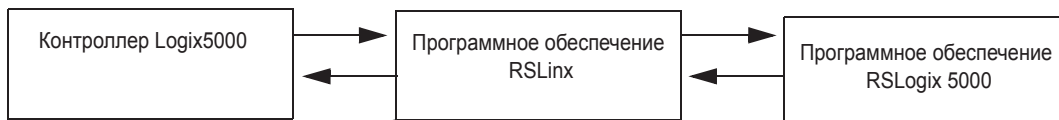
3. Поставьте или снимите флажок.

4. Выберите 



## Настройка связи с контроллером через последовательный порт

Программное обеспечение RSLinx® обеспечивает связь между контроллерами Logix5000 и вашим приложением, например, таким как RSLogix 5000. Для организации связи с контроллером (т.е. загрузки и контроля данных) сконфигурируйте соответствующим образом RSLinx.



**драйвер (driver)** - устанавливает связь по заданной сети.

**путь (path)** - коммуникационный путь к устройству. Для того чтобы определить путь необходимо раскрыть драйвер и выбрать устройство.

Используйте кабель последовательного порта для установки связи «точка-точка» между последовательными портами компьютера и контроллера.

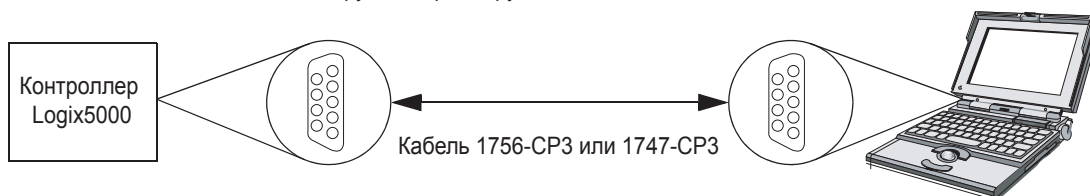
### ПРЕДУПРЕЖДЕНИЕ



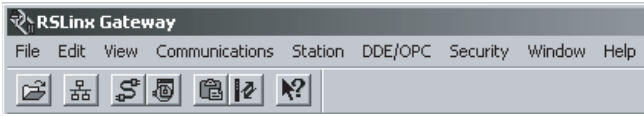
Если вы подключаете или отключаете кабель последовательного порта под напряжением от модуля или устройства на другом конце, то может возникнуть электрическая дуга. Это может вызвать взрыв в случае работы в опасных условиях.

Перед такой операцией убедитесь, что питание отключено или отсутствуют опасные условия.

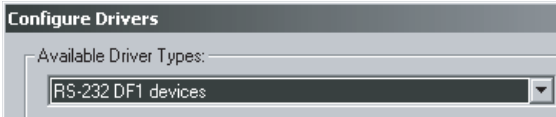
1. Подключите кабель к компьютеру и контроллеру.



2. Конфигурирование драйвера RS-232:

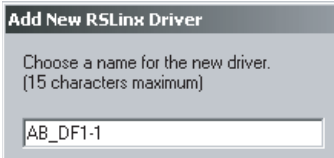


a. Запустите программное обеспечение RSLinx.

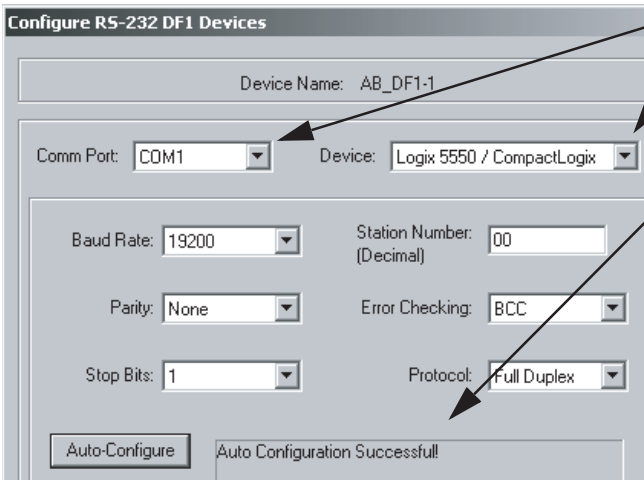


b. Щелкните на  .

c. Выберите *RS-232 DF1 devices* (драйверы RS-232 DF1) и затем выберите  (Добавить новый)

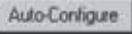


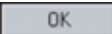
d. Согласитесь с именем по умолчанию.

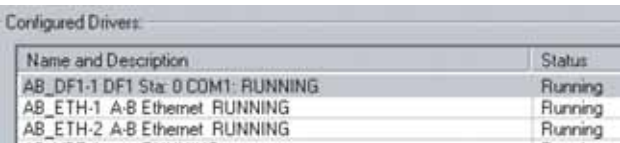


e. Выберите COM порт вашего компьютера.

f. Выберите *Logix 5550/CompactLogix*.

g. Выберите  (Автоматическое конфигурирование)

h. Когда закончится автоматическое конфигурирование, выберите  .



← Драйвер успешно сконфигурирован и запущен.

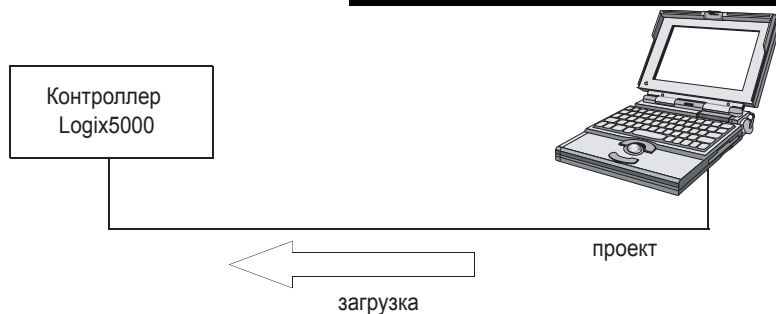
## Загрузка проекта в контроллер

Для того чтобы проект выполнялся в контроллере, его необходимо загрузить в контроллер.

### ВНИМАНИЕ



Когда вы загружаете проект или обновляете микропрограммное обеспечение, все активные сервоприводы выключаются. Перед загрузкой проекта или обновлением микропрограммного обеспечения убедитесь, что это *не приведет* к неожиданному движению приводов.



**загрузка** - перенос проекта из компьютера в контроллер для его выполнения.

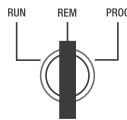
- Когда вы загружаете проект, вы теряете тот проект и данные, которые уже загружены в контроллер в настоящий момент (если таковой имеется).
- Если версия контроллера не совпадает с версией проекта, вам будет предложено обновить версию микропрограммного обеспечения контроллера. Программное обеспечение RSLogix 5000 позволяет вам обновлять микропрограммное обеспечение контроллера в качестве одной из операций при загрузке.

### ВАЖНО

Для обновления микропрограммного обеспечения, во-первых, установите инструментарий для обновления.

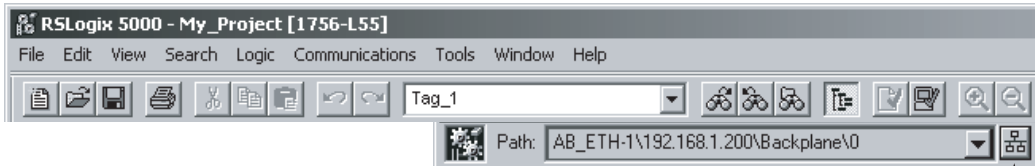
- Этот инструментарий поставляется на CD вместе с программным обеспечением RSLogix 5000.
- Для загрузки инструментария обратитесь на сайт [www.ab.com](http://www.ab.com). Выберите *Product Support* (Поддержка продукта). Выберите *Firmware Updates* (Обновление микропрограммного обеспечения).

1. Установите переключатель контроллера в положение:



2. Определите путь к контроллеру:

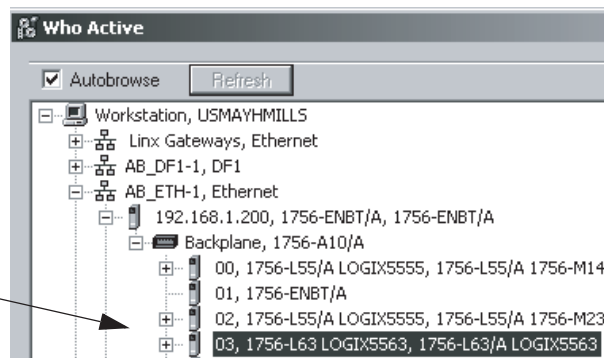
a. Откройте тот проект RSLogix 5000, который вы хотите загрузить.



b. щелкните на  .

c. откройте контроллер.

- Чтобы открыть соответствующий уровень, щелкните на значке + .
- Когда вы увидите нужный контроллер, выберите его.



3. Загрузка проекта:

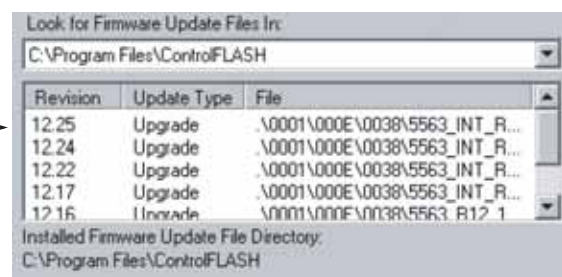
a. выберите  .

Какой отклик дает RSLogix 5000? Ошибка при загрузке в контроллер. Версии проекта и микропрограммного обеспечения контроллера не совместимы.

Загрузка в контроллер

b. Выберите  .

b. Выберите  .



c. выберите версию для контроллера.

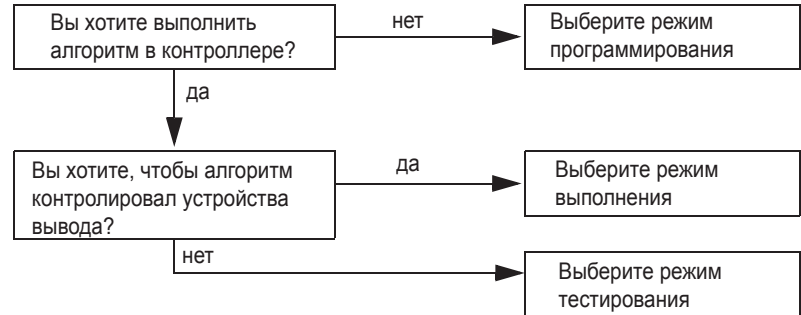
d. Выберите  и затем  .

b. Выберите .

## Выбор рабочего режима контроллера

Для выполнения или останова выполнения алгоритма в контроллере, измените рабочий режим контроллера.

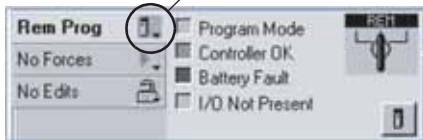
1. Определите, какой режим вам необходим:



2. Установите переключатель в положение **RUN REM PROG**



3. Войдите в режим он-лайн с контроллером.



4. Выберите режим.

**Для заметок:**

---

## Организация проекта

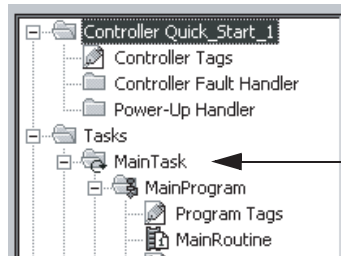
### Использование этой главы

В этой главе представлена более подробная информация об организации трассировки программы и структур данных для контроллера:

Если вы хотите:	См. стр.
Сконфигурировать выполнение задачи	2-2
Создать дополнительные программы	2-3
Создать пользовательские типы данных	2-5
Определить процедуры	2-7
Присвоить главные процедуры	2-10
Сконфигурировать контроллер	2-11
Сконфигурировать модули ввода/вывода	2-12

## Конфигурирование выполнения задачи

Новый проект содержит задачу «по умолчанию» для выполнения вашего алгоритма.



**Задача (task)** - определяет плановую и приоритетную информацию, необходимую для выполнения (сканирования) вашего алгоритма.

В данном руководстве мы ограничили проект одной задачей, с одним из следующих типов выполнения:

Если вы хотите выполнять вашу логику:	Сконфигурируйте задачу с данным типом выполнения:
<p>Все время</p> <p>Выполнение логики</p> <p>Задача автоматически перезапускается    Задача автоматически перезапускается    Задача автоматически перезапускается    Задача автоматически перезапускается</p>	<p>Непрерывный</p> <p>Это настройка по умолчанию для <i>MainTask</i>.</p>
<p>В заданный период</p> <p>Выполнение логики</p> <p>Задача завершилась    Период истекает    Задача перезапускается    Задача завершилась    Период истекает    Задача перезапускается</p>	<p>Периодический</p> <p>Вы задаете период, в течение которого выполняется задача</p>

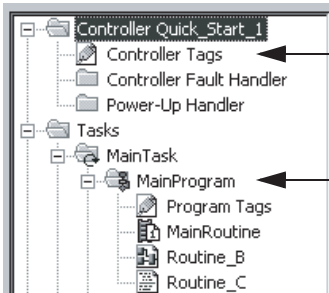
- Щелкните правой клавишей мыши и выберите *Properties (Свойства)*.
- Щелкните на закладке *Configuration (Конфигурация)*.
- Выберите *Periodic (Периодический)*.
- Введите с клавиатуры период для данной задачи.
- Выберите **OK**.

При использовании нескольких задач или при выполнении задачи по заданному событию (триггер), см. публикацию «Контроллеры Logix500, Общая методика», 1756-PM001.



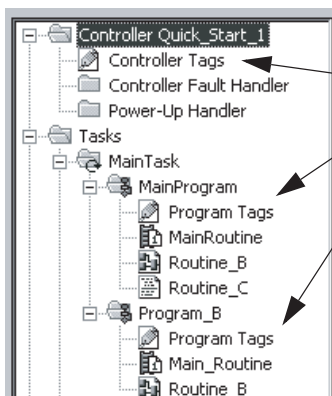
## Создание дополнительных программ

Контроллер Logix5000 позволяет делить приложение на несколько программ, каждая из которых имеет собственные теги (данные).



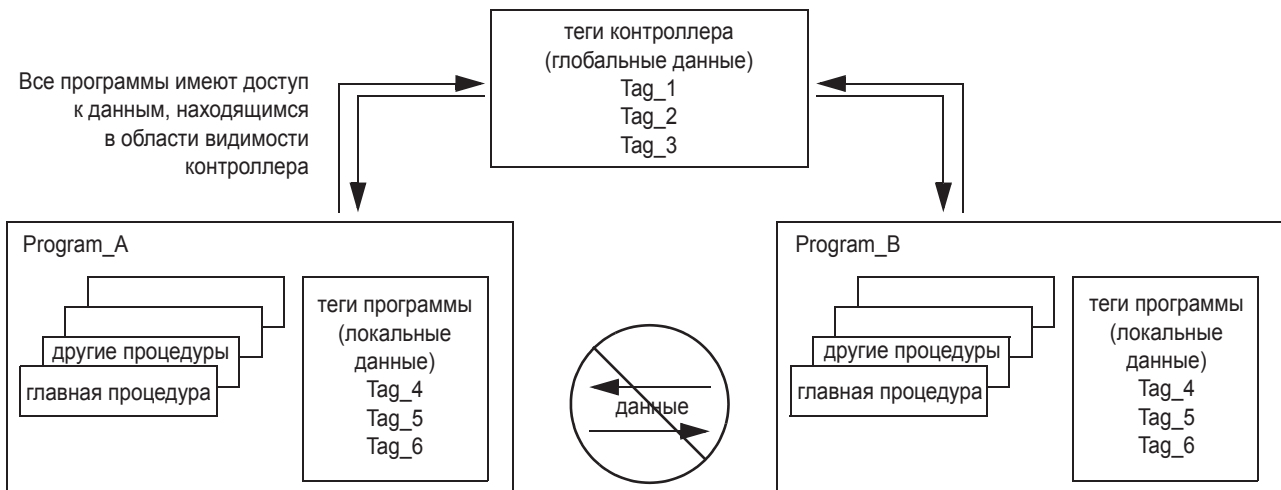
**Тег (tag)** - хранилище данных. Не существует фиксированной таблицы или цифрового формата для адресов данных. Имя тега является адресом (нет перекрестной ссылки к физическому адресу). Вы создаете те теги, которые хотите использовать.

**Программа (program)** - программа и данные, изолированные от других программ и данных. Каждая программа содержит одну или более процедур как сопряженные данные.



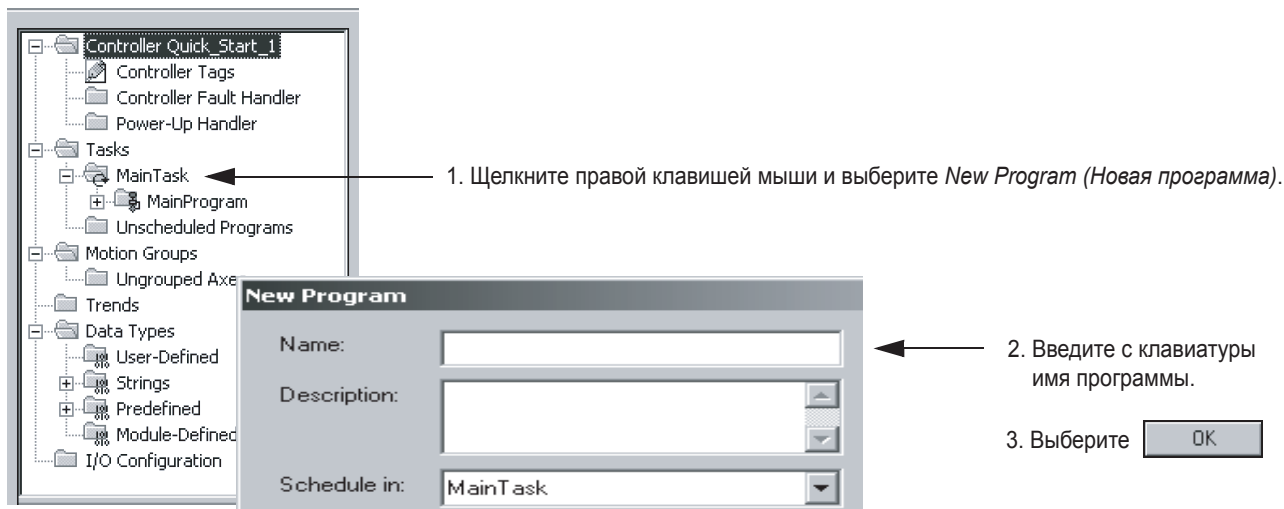
**Область видимости (scope)** - определяет, является ли тег доступным всем программам (тег контроллера) или ограничивается заданной программой (тег программы). Данные в пределах видимости программы изолированы от других программ.

Нет необходимости отслеживать конфликты имен тегов между программами.



Данные в области видимости программы изолированы от других программ:

- Процедуры не могут обращаться к данным, находящимся в области видимости другой программы.
- Вы можете многократно использовать в нескольких программах имя тега, находящегося в области видимости программы. Например, обе программы (Program\_A и Program\_B) могут иметь тег программы с именем Tag\_4.



**СОВЕТ**

Имена:

- должны содержать только буквенные знаки, цифровые знаки и знаки подчеркивания ( \_ )
- должны начинаться с буквенного знака или знака подчеркивания
- должны содержать не более 40 знаков
- знаки подчеркивания не должны располагаться подряд или в конце строки
- не чувствительны к регистру

Определенные теги должны находиться в видимости контроллера.

Если вы хотите использовать тег:	То используйте данную область видимости:
Более чем в одной программе в рамках проекта	Теги контроллера
В инструкции Message (MSG) (Сообщение)	
Для производства или потребления тега	
Для связи с терминалом PanelView	
Только в одной программе	Теги программы для данной программы

## Создание пользовательского типа данных

Пользовательский тип данных позволяет вам организовывать данные в соответствии с вашей машиной или процессом. Это упрощает разработку программы и позволяет создавать самодокументирующиеся коды, которые просты для обслуживания.

**тег** - хранилище данных. Не существует фиксированной таблицы или цифрового формата для адресов данных. Имя тега является адресом. Вы создаете те теги, которые хотите использовать.

**тип данных** - определяет тип данных, которые сохраняются в теге, например, двоичный, целый, с плавающей точкой, строковый и т.д.

**массив** - задает блок данных (файл). В одном блоке используется один тип данных. Массив может иметь размерность 1, 2 или 3.

**структура** - объединяет типы данных в повторно используемый формат (шаблон для тегов). Используйте структуру как основу для нескольких тегов с одинаковой компоновкой данных.

**член** - описывает отдельный участок данных в пределах структуры.

**пользовательский тип данных** - создает вашу собственную структуру, которая имитирует устройства. Пользовательский тип данных хранит все данные, относящиеся к конкретным особенностям вашей системы. Это позволяет хранить родственные данные вместе и упрощает их локализацию независимо от их типа.

Tag Name	Type
Conveyor_Direction	BOOL
[-]Conveyor_Speeds	DINT[8]
[-]Motor_Start_Delay	TIMER
[-]MyData_1	DINT
[-]MyData_2	DINT
MyData_3	REAL
[-]MyData_4	DINT
MyData_5	REAL
[-]MyData_6	STRING
[-]Tank_1	Tank

Name	Data Type
PRE	DINT
ACC	DINT
EN	BOOL
TT	BOOL
DN	BOOL

Name	Style
Level	Decimal
Pressure	Decimal
Temp	Float
Agistator_Speed	Decimal
Ingredient_A	Decimal
Ingredient_B	Decimal

При создании пользовательского типа данных помните о следующем:

Указание:	Подробности:
1 Используйте сквозные описания.	См. «Описание пользовательского типа данных» на стр. 4-2.
2 Данные, которые представляют устройства ввода вывода, требуют дополнительного программирования.	Если вы включаете члены, представляющие устройства ввода/вывода, вы должны использовать логику для копирования данных между членами в пользовательском типе данных и соответствующими тегами ввода/вывода.
3 Если вы включаете массив в качестве члена, ограничьте массив одним измерением.	Многомерные массивы <i>не</i> допускаются в пользовательском типе данных.
4 При использовании типов данных BOOL, SINT или INT размещайте члены одного типа данных друг за другом:	Контроллеры Logix5000 распределяют память в 4-х байтовых участках. Если вы объединяете в последовательность более короткие данные, контроллер их упакует столько, сколько умещается в 4-х байтовом участке.

**более эффективно**

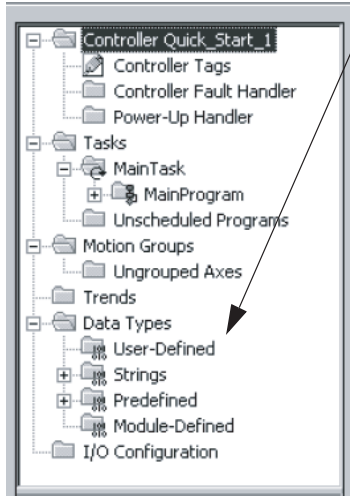
BOOL
BOOL
BOOL
DINT
DINT

**менее эффективно**

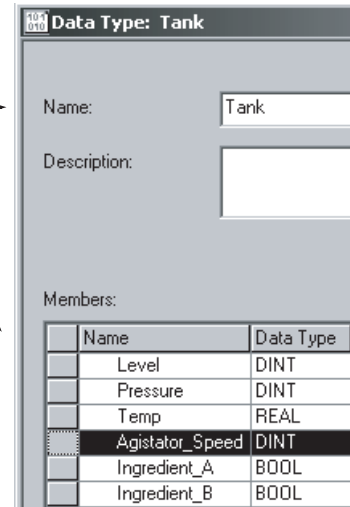
BOOL
DINT
BOOL
DINT
BOOL

Для создания пользовательского типа данных и тегов, которые их используют:

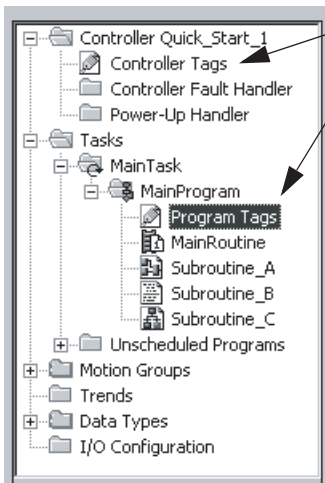
1. Создайте пользовательский тип данных:



- a. Щелкните правой клавишей мыши и выберите *New Data Type* (Новый тип данных).
- b. Введите с клавиатуры имя для этого типа данных (а не имя тега, который будет использовать этот тип данных).
- c. Введите члены. В качестве необязательной операции введите описания для каждого члена.
- d. Выберите



2. Создание тега, который использует пользовательский тип данных:



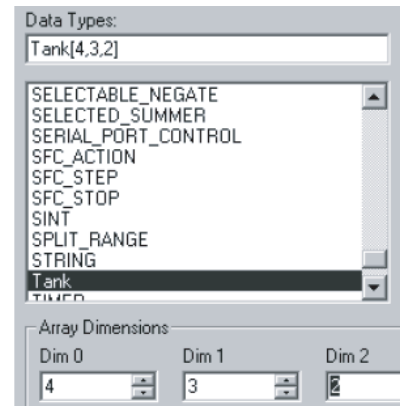
- a. Щелкните правой клавишей мыши на области видимости, нужной для тега и выберите *Edit Tags* (Редактирование тегов).

Tag Name	Alias For	Base Tag	Type
+MyData_4			DINT
MyData_5			REAL
+MyData_6			STRING
Tank_1			Tank ...

- b. Введите с клавиатуры имя для этого тега.
- c. Введите с клавиатуры имя пользовательского типа данных из пункта 1.

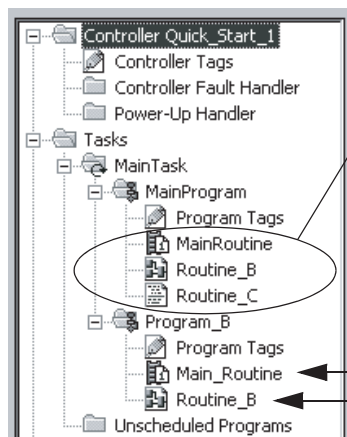
3. Если вы хотите, чтобы тег представлял собой массив (многочисленные экземпляры этого типа данных):

- c. Выберите тип данных и щелкните на
- d. Задайте размерность массива.
- e. Выберите



## Задание процедур

Если ваш проект уже имеет требующиеся программы, вам необходимо определить и создать процедуры для каждой программы.



**процедура** - обеспечивает исполняемый код (алгоритм) для программы (подобно программному файлу в контроллере PLC или SLC).

**главная процедура** - каждой программе вы приписываете главную процедуру.

- Когда программа выполняется, автоматически выполняется ее главная процедура.
- Используйте главную процедуру для управления выполнением других процедур в подпрограмме.
- Для вызова (выполнения) другой процедуры (подпрограммы) из программы, используйте инструкцию Jump to Subroutine (JSR) (переход к подпрограмме).

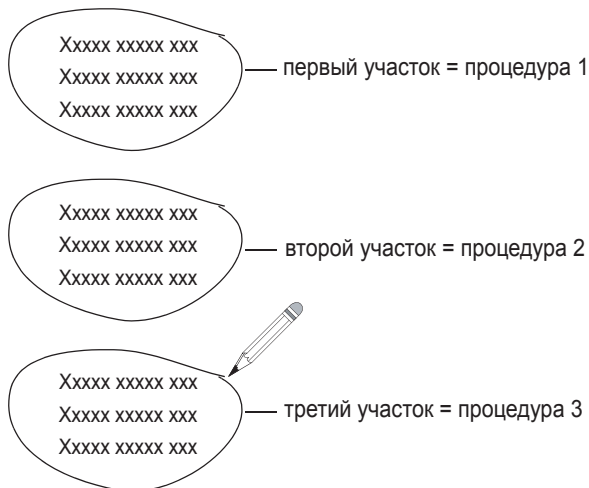
**подпрограмма** - всякая процедура, не являющаяся главной процедурой и процедурой обработки ошибок. Для выполнения подпрограммы используйте инструкцию Jump to Subroutine (JSR) в другой процедуре, например, в главной процедуре.

## Задание процедуры для каждого участка вашей установки или процесса

Для упрощения разработки, тестирования и отладки вашего проекта разделите его на процедуры (подпрограммы).

1. Выделите каждый физический участок вашей установки или процесса.
2. Назначьте процедуру каждому из таких участков.

### Описание вашей установки или процесса



## Определение установленных языков программирования

Чтобы определить, какие языки программирования установлены в вашей версии программного продукта RSLogix 5000:

1. Запустите RSLogix 5000.
2. В меню *Help* (Справка) выберите *About RSLogix 5000* (О RSLogix5000).

За инструкциями по добавлению языка программирования обращайтесь к публикации 1756/SG001 «Руководство по выбору *ControlLogix*».

## Выбор языка программирования для каждой процедуры

Выберите подходящий язык программирования для каждой процедуры:

- Контроллеры Logix5000 позволяют использовать следующие языки:
  - релейная логика
  - функциональная блок-схема
  - последовательная функциональная схема
  - структурированный текст
- Используйте любую комбинацию этих языков в одном проекте.

Как правило, если участок вашего кода представляет собой:	Используется этот язык:
непрерывное или параллельное выполнение нескольких операций (не последовательных)	релейная логика
булевы или побитовые операции	
сложные логические операции	
обработку сообщений и передачу данных	
блокировку механизма	
операции, которые при необходимости должны интерпретироваться обслуживающим персоналом для диагностики установки или процесса	
управление непрерывным процессом и приводом	функциональная блок-схема
управление циклом	
расчеты в цикле	
управление несколькими операциями высокого уровня	последовательная функциональная схема
повторяемые последовательности операций	
групповую обработку	
управление перемещением с помощью структурированного текста	
операции конечного автомата	
сложные математические операции	структурированный текст
обработку специального массива или таблицы в цикле	
обработку строк ASCII или протоколов	

## Разбивка каждой процедуры на более осмысленные части

Если процедура использует этот язык:	То:	Пример:
релейная логика  структурированный текст	Разбейте большие процедуры на несколько меньших	<p>Для непрерывного выполнения нескольких сложных булевых операций...</p> <p>... создайте отдельную процедуру для каждой операции.</p>
функциональная блок-схема (ФБС)	Внутри процедуры ФБС создайте отдельный лист для каждого функционального цикла каждого устройства (двигателя, клапана и т.п.)	<p>Для управления 4 клапанами, при котором каждому клапану требуется обратная связь с подтверждением, что он находится в заданном положении, ...</p> <p>... создайте отдельный лист для каждого клапана.</p>
последовательная функциональная схема (ПФС)	Разбейте ПФС на шаги	<p>Для выполнения следующей последовательности:</p> <ol style="list-style-type: none"> <li>1 Заполнение резервуара</li> <li>2 Смешивание ингредиентов в резервуаре.</li> <li>3 Опорожнение резервуара...</li> </ol> <p>... сделайте каждый этап (заполнение, смешивание, опорожнение) отдельным шагом.</p>

## Определение главных процедур

Для каждой программы требуется главная процедура. После создания процедур, определите главную процедуру для каждой программы.

### ВАЖНО

В проекте «по умолчанию», *MainProgram* (Головная программа) уже имеет главную процедуру (*MainRoutine*). Вам необходимо определить главную процедуру только для каждой из дополнительных программ, которые вы создаете.

Для создания главной процедуры:

1. Щелкните правой клавишей мыши и выберите *Properties* (Свойства).

2. Щелкните на закладке *Configuration* (Конфигурация).

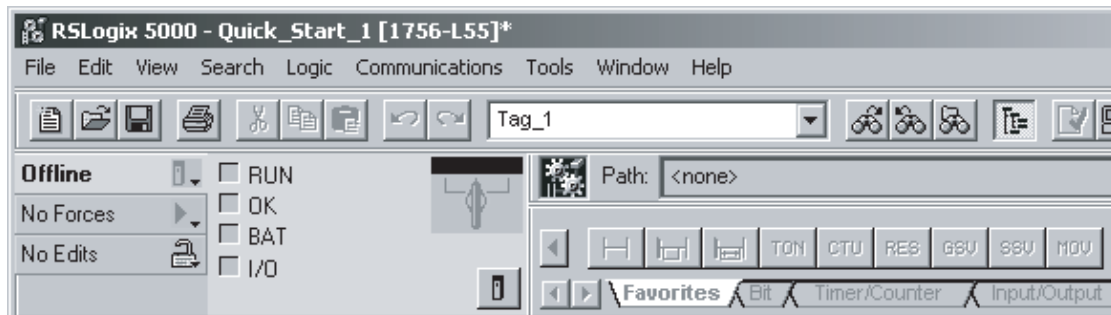
3. Выберите главную процедуру.

4. Выберите

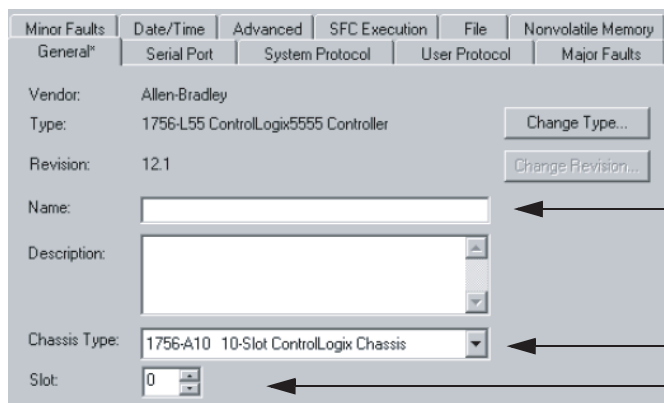


## Конфигурирование контроллера

Если вы хотите изменить конфигурацию контроллера, например, имя, размер шасси или номер слота, используйте диалоговое окно Controller Properties (Свойства контроллера).



1. Щелкните на кнопке Controller Properties (Свойства контроллера)
2. Измените нужные параметры (пункты, относящиеся только к этому контроллеру).

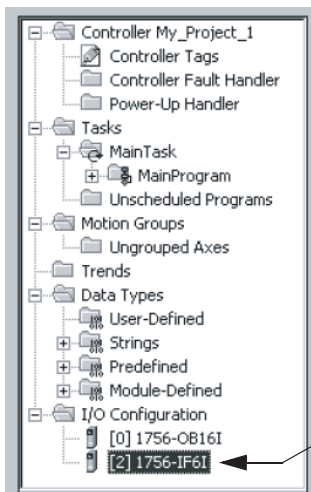


- e. тип контроллера
- f. имя контроллера
- g. размер шасси для контроллера
- h. номер слота контроллера

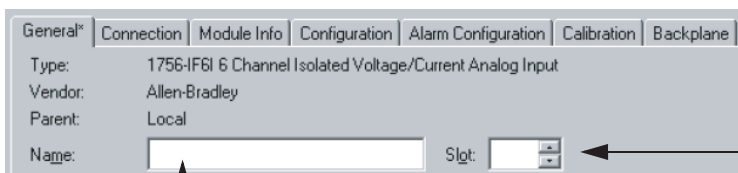
3. Выберите

## Конфигурирование ввода/вывода

Для изменения работы модуля, используйте окно Module Properties (Свойства модуля). Опции конфигурирования меняются от модуля к модулю.



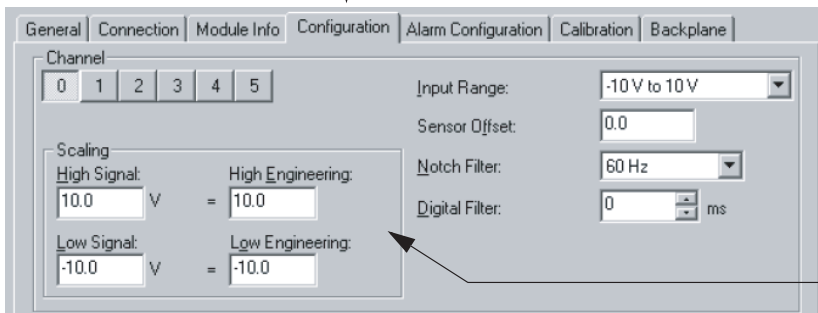
1. Щелкните правой клавишей мыши на модуле и выберите *Properties (Свойства)*.
2. Для изменения имени или номера слота, используйте закладку *General (Общие)*.



положение модуля на шасси или шине

имя модуля

3. Для изменения конфигурации, щелкните на закладке *Configuration (Конфигурация)*. Некоторые модули имеют несколько закладок для конфигурирования.



диапазон

масштаб

## Программирование проекта в режиме оффлайн

### Использование этой главы

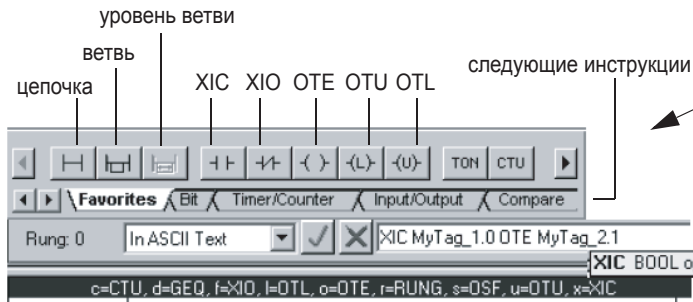
В данной главе представлена подробная информация о программировании логики для процедуры и создании тегов для этой логики.

<b>Если вы хотите:</b>	<b>См. стр.</b>
Ввести релейную логику	3-2
Экспортировать/импортировать релейную логику	3-6
Ввести функциональную блок-схему	3-9
Использовать разметочную панель для функционального блока	3-12
Ввести структурированный текст	3-14
Ввести последовательную функциональную схему	3-16
Присвоить операнды	3-18
Проверить проект	3-20
Просмотреть правила для тегов	3-22

В этой главе рассказывается как программировать проект в режиме оффлайн. Программирование в режиме онлайн требует дополнительных операций, описанных в главе 6 «Программирование в режиме онлайн».

## Ввод релейной логики

Для ввода релейной логики вам понадобятся следующие опции:

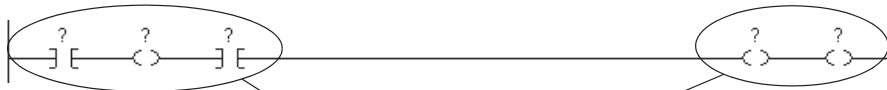


**Буксировка элементов логики** - Используйте инструмент "Элементы языка" (Language Element) для буксировки этих элементов в цепочку.

**Текст ASCII** - Используйте текст ASCII для ввода или редактирования логики. Подсказка поможет вам ввести требуемые операнды. Текст ASCII используется обычно в следующем формате:

Mnemonic operand\_1 operand\_2

**Клавиши быстрого вызова** - присвойте какой-либо логический элемент (цепочка, ветвь, инструкция) клавише клавиатуры. Для того чтобы добавить этот элемент справа или ниже курсора, нажмите клавишу, приписанную этому элементу.



**Ряд выводов** - разместите инструкции нескольких выводов последовательно на цепочке.

**Инструкции чередующегося ввода/вывода** - последняя инструкция цепочки должна быть инструкцией вывода.



**Параллельные ветви** - нет ограничений на количество параллельных ветвей в цепочке (до 6 уровней вложений).

**Не идентифицированные операнды** - введите логику без определения операндов. Программное обеспечение RSLogix 5000 позволяет вам ввести и сохранить логику без присваивания операндов. Это позволяет разрабатывать алгоритм в режиме итераций и сохранить библиотеки кодов для повторного использования.

## Буксировка элемента

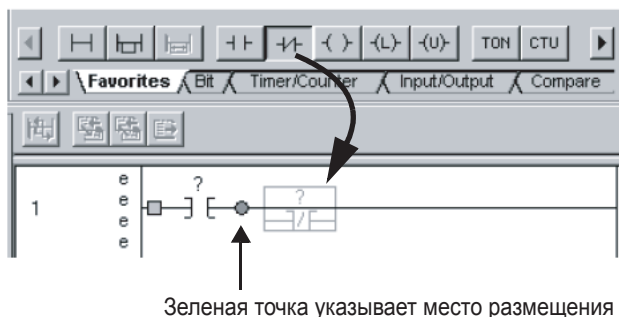
**Чтобы:**

**Сделайте следующее:**

добавить цепочку

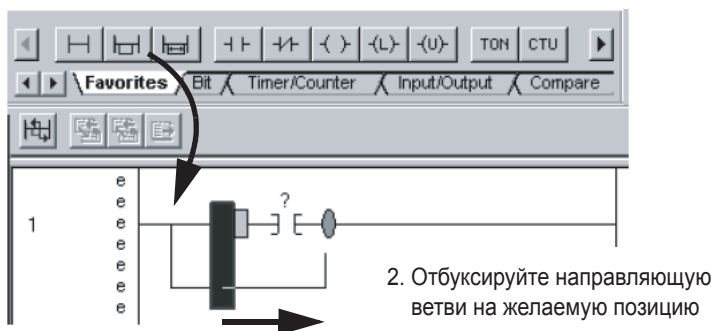
Отбуксируйте кнопку для цепочки или инструкцию прямо на нужное место.

добавить инструкцию

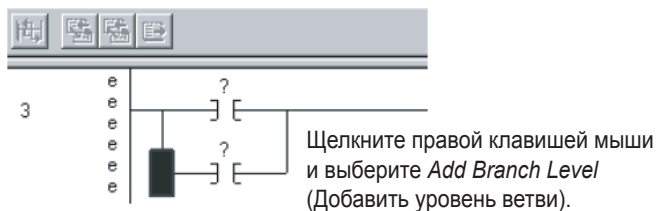


добавить ветвь

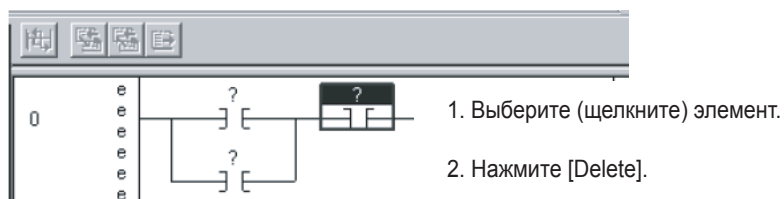
1 Отбуксируйте эту кнопку туда, где начинается ветвь. Зеленая точка указывает место размещения.



добавить уровень ветви

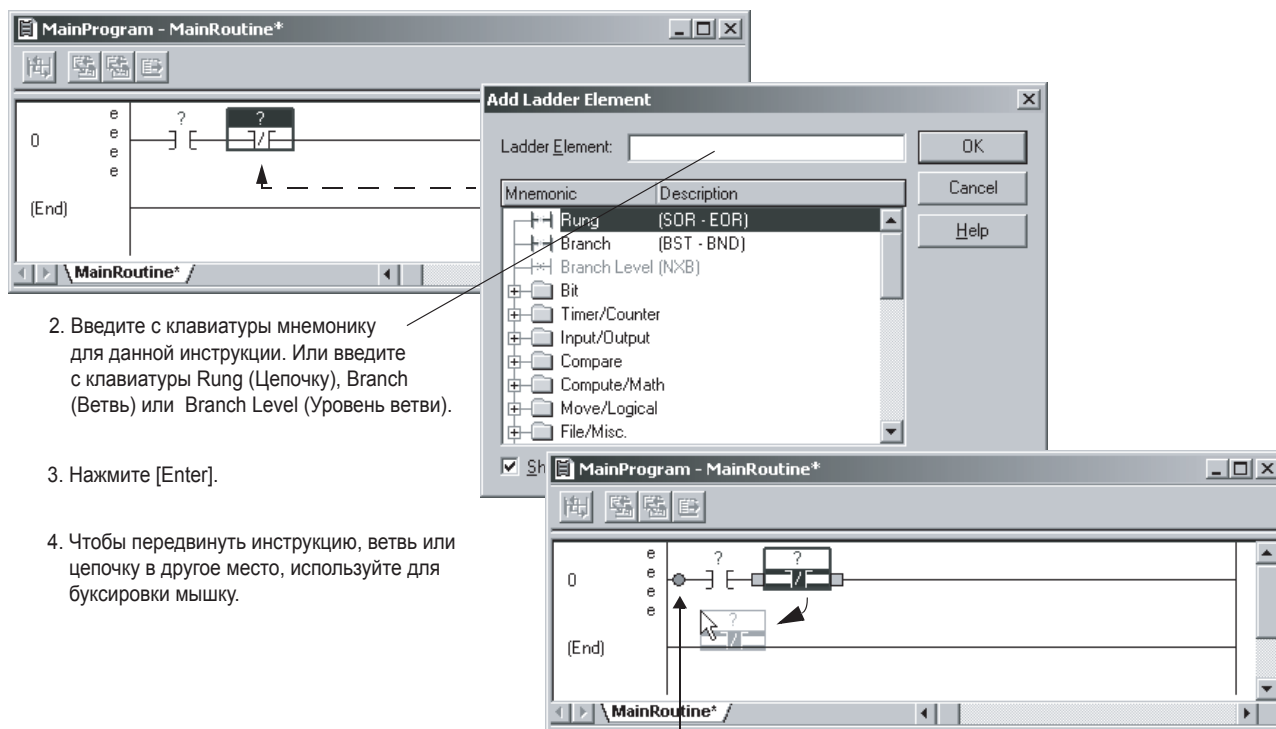


уничтожить элемент



## Использование клавиатуры для добавления элемента

1. Нажмите [Insert].



2. Введите с клавиатуры мнемонику для данной инструкции. Или введите с клавиатуры Rung (Цепочку), Branch (Ветвь) или Branch Level (Уровень ветви).

3. Нажмите [Enter].

4. Чтобы передвинуть инструкцию, ветвь или цепочку в другое место, используйте для буксировки мышку.

Зеленая точка указывает положение (точка сброса).

## Ввод логики с использованием текста ASCII

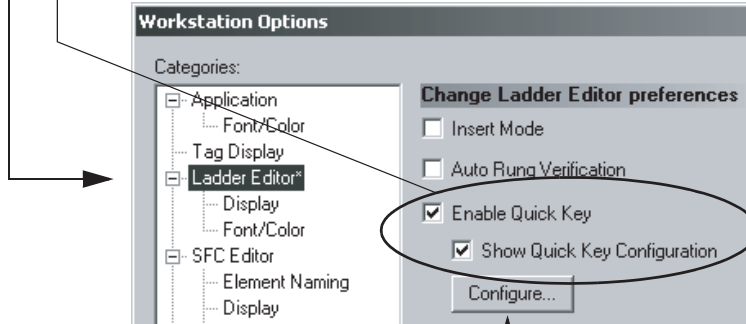
1. Дважды щелкните на цепочке.



2. Введите для цепочки текст ASCII.

## Использование клавиш быстрого ввода


1. Выберите Tools (Инструменты) => Options (Опции).
2. Выберите (щелкните) Ladder Editor (Редактор логики).
3. Выберите (поставьте) эти флажки.

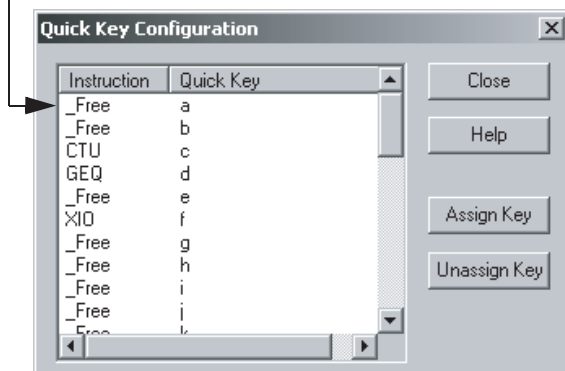


4. Присвойте клавишу элементу:

a. Выберите 

b. Для желаемой клавиши выберите нужный элемент

c. Когда вы присвоили желаемые клавиши, выберите 

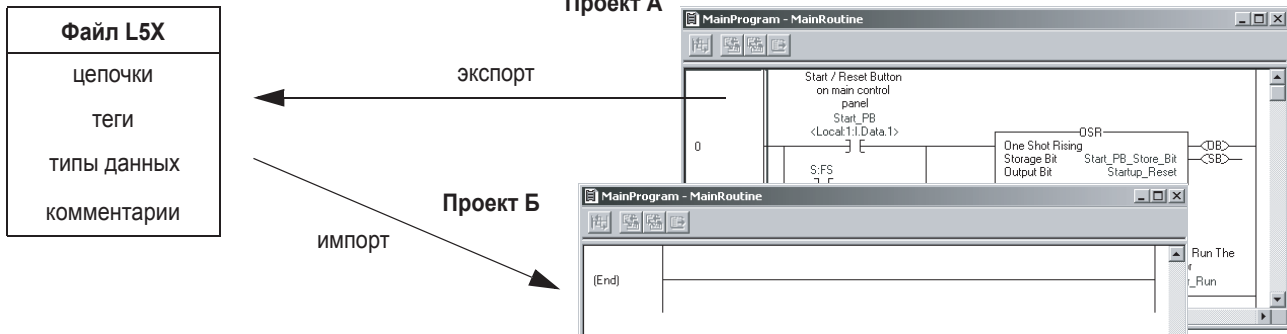


## Экспорт/импорт релейной логики



RSLogix 5000 версии 13.0 или выше

Если вы хотите повторно использовать релейную логику из другого проекта, просто экспортируйте логику в файл L5X и импортируйте его в требуемый проект. Файл L5X содержит все необходимое для логики, за исключением модулей ввода/вывода.



### Когда вы импортируете цепочки...

Когда вы импортируете цепочки, программное обеспечение RSLogix 5000 демонстрирует список тегов и пользовательских типов данных, которые сопровождают данные цепочки. Используйте этот список для управления типами данных, которые создаются во время операции импорта.

Колонка *Operation* (Операция) показывает, что произойдет с каждым тегом и типом данных при импорте. Программное обеспечение либо создает их, либо использует существующие в данном проекте, либо отменяет их (не импортирует).

Если нужно, вы можете переименовать тег для лучшего соответствия проекту.

Если вы размещаете переменные для цепочек в пользовательском формате, у вас меньше тегов для управления.

Если тег уже существует в проекте, вы можете:

- Использовать существующий тег, который отменит тег в библиотечном файле и привяжет логику к существующему тегу.
- Переименовать тег, который создает новый.

The screenshot shows the 'Import Configuration' dialog box with the 'Tags' tab selected. It contains a table with columns: Tag Name, Alias For, Type, Description, and Operation. The 'Conveyor\_Type' tag is highlighted with a red circle. The 'Operation' column shows 'Create New' for 'Conveyor\_Type' and 'Use Existing' for 'Local:2:0'.

Tag Name	Alias For	Type	Description	Operation
Conveyor_Type		Conveyor_Type	Conveyor CN1	Create New
CN2_M	Local:2:0.Data.0		Conveyor CN1 Motor	Create New
Estop_Disabled		BOOL	No Estop pressed	Use Existing
Local:1:I		AB:1756_DI:1:0		Discard
Local:2:O		AB:1756_DO:0:0		Use Existing

Не создается новых тегов ввода/вывода.

Если тег ввода/вывода уже существует в данном проекте, операция импортирования использует этот тег для любых псевдонимов этого имени тега. При импортировании убедитесь, что вы пометили теги псевдонимы.

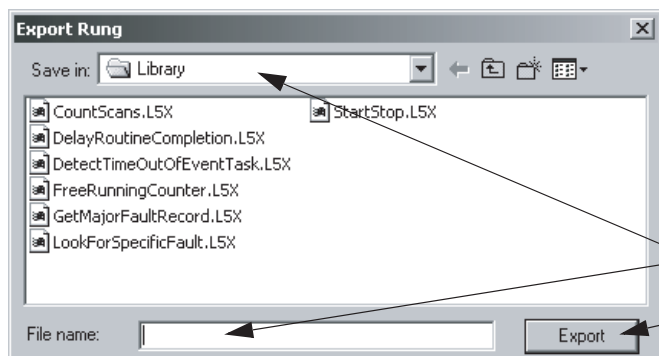
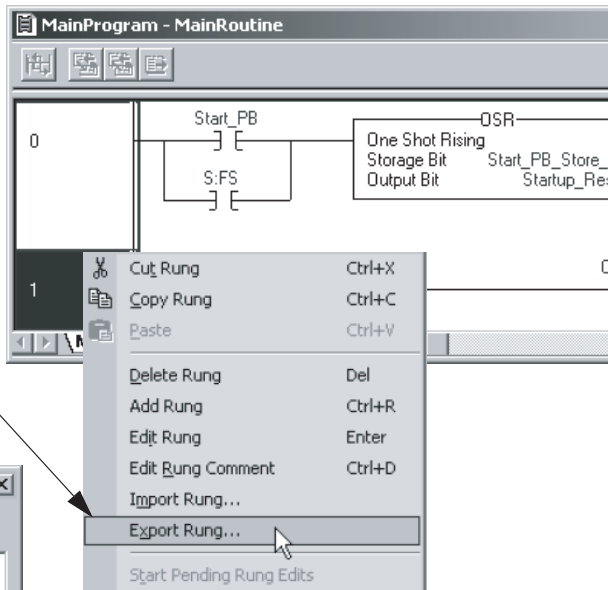


### Экспорт цепочек

1. Выберите цепочки для экспорта:

Если цепочки:	To:
в последовательности	Щелкните на первой цепочке, а затем нажмите [Shift] + щелчок на последней цепочке
вне последовательности	Щелкните на первой цепочке, а затем нажмите [Ctrl] + щелчок на каждой дополнительной цепочке

2. Щелкните правой клавишей мыши на вашем выборе и выберите *Export Rung* (экспортировать цепочку).



3. Выберите расположение и имя для файла.

4. Создайте файл.

## Импорт цепочек

1. Щелкните правой клавишей мыши на расположении цепочек и выберите *Import Rung* (Импортировать цепочку).

2. Выберите файл для импортирования.

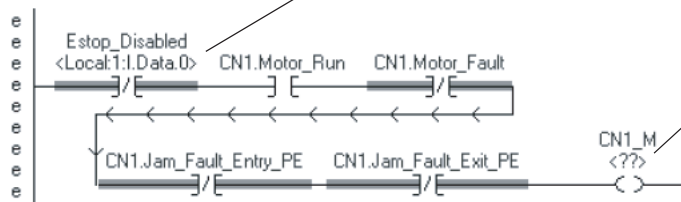
3. Проверьте конфликтность имен.

4. Импортируйте файл.

Tags	Data Types	Tag Name	Alias For	Type	Description
		CN2		Conveyor_Type	Conveyor C
		CN2_M	Local:2:0.Data.0		Conveyor C
		Estop_Disabled		BOOL	No Estop p
		Local:1:I		AB:1756_DI:1:0	
		Local:2:O		AB:1756_DO:0:0	

## Проверка тегов-псевдонимов

цепочки, которые вы импортируете

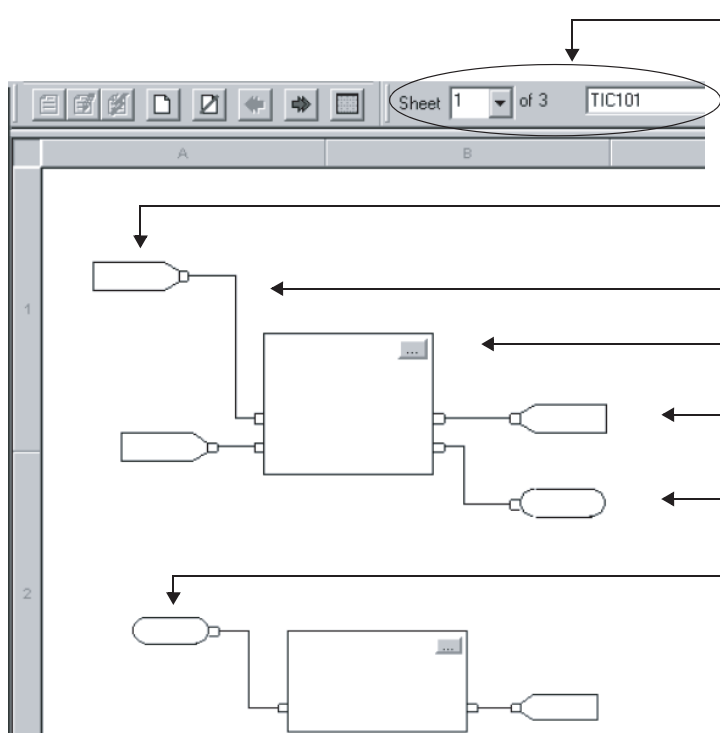


Если вы импортируете теги псевдонимы, убедитесь, что они указывают на правильные базовые теги. Если тег является псевдонимом для тега, который уже существует в проекте, программное обеспечение настраивает взаимоотношения псевдонимов и базовых тегов.

Если проект не имеет базового тега, вы можете либо создать базовый тег, либо указать псевдоним для другого базового тега.

## Ввод функциональной блок-схемы

Функциональная схема позволяет визуально определять поток данных между инструкциями. Этот поток данных затем определяет порядок выполнения инструкций.



**лист (sheet)** - делит схему на участки без влияния на выполнение. (При выполнении процедуры выполняются все листы)

**входная ссылка (input reference (IREF))** - считывает значение из тега или устанавливает постоянное значение

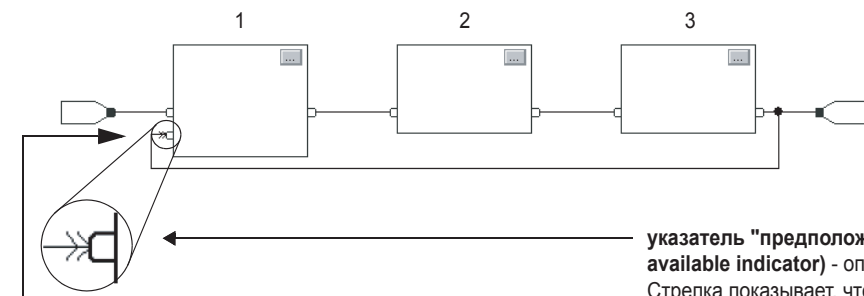
**связь (wire)** - передает данные между элементами

**инструкция (instruction)** - производит выходные значения на основе входных значений

**выходная ссылка (output reference (OREF))** - записывает значение в тег

**коннектор выходной связи (output wire connector (OCON))** - записывает значения в один или более ICON

**коннектор входной связи input wire connector (ICON)** - считывает значение из OCON на том же самом листе или на другом листе в процедуре. Чтобы считать значение из другой процедуры, используйте комбинацию OREF/IREF.



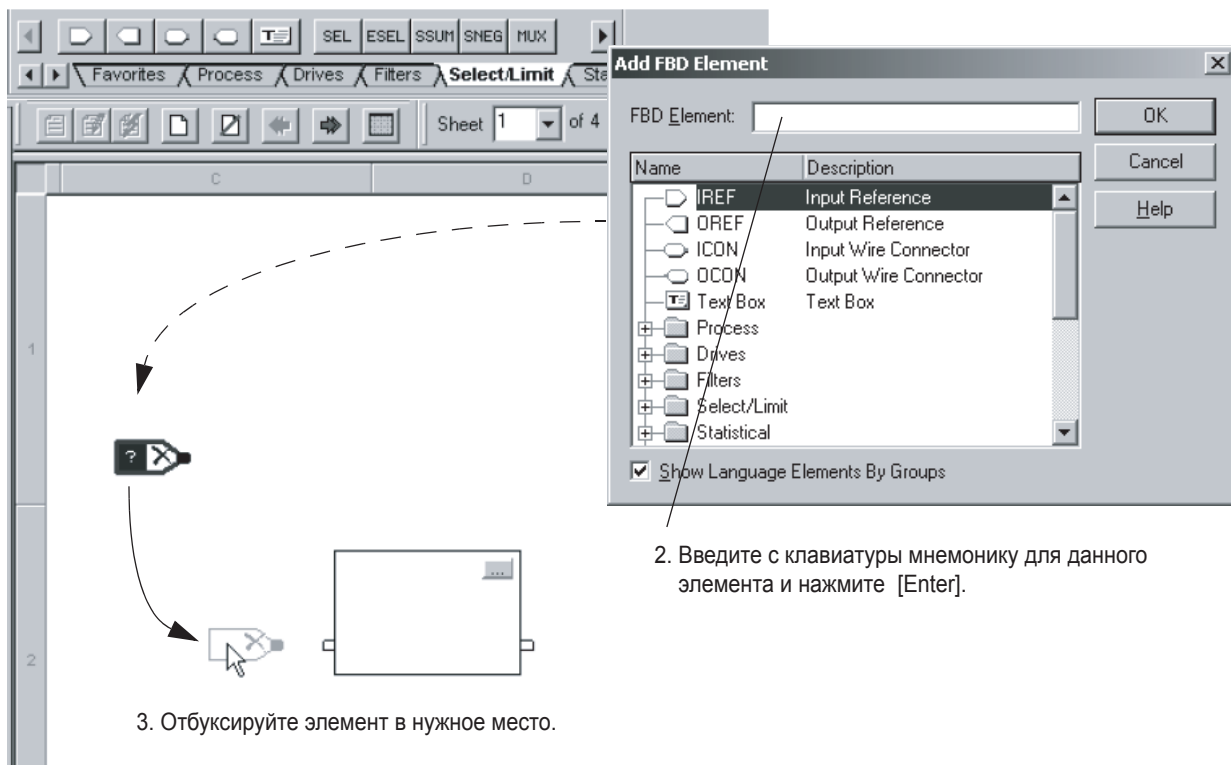
Эта точка входа использует выход, который произвел блок 3 на предыдущем сканировании.

**указатель "предположение о наличии данных" (assume data available indicator)** - определяет поток данных в пределах цикла. Стрелка показывает, что данные служат в качестве входа для первого блока в цикле.

Если группа блоков составляет цикл, вам необходимо определить, какой блок выполняется первым. Используйте указатель Assume Data Available для того, чтобы пометить входную связь, которая создает цикл (обратная связь).

## Использование клавиатуры для добавления элемента

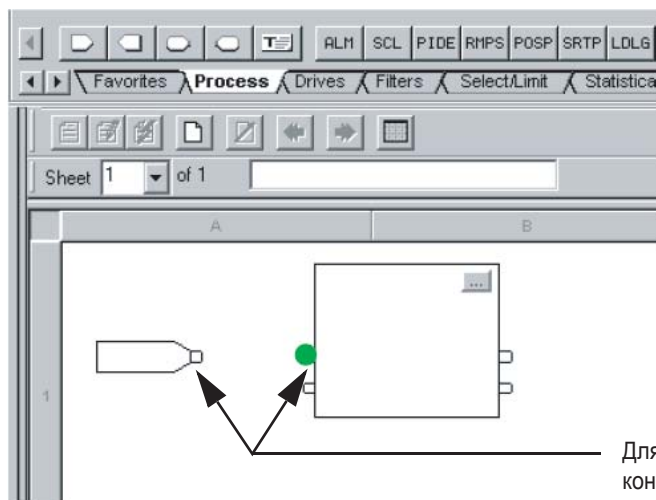
1. Нажмите [Insert].



2. Введите с клавиатуры мнемонику для данного элемента и нажмите [Enter].

3. Отбуксируйте элемент в нужное место.

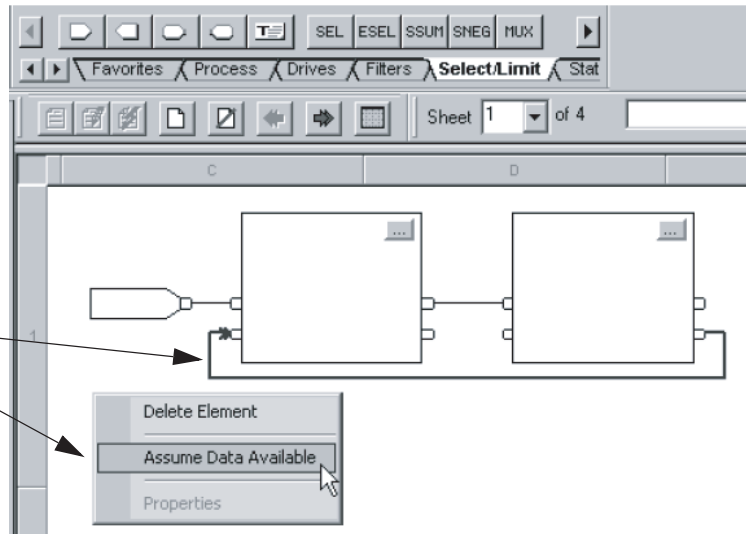
## Связь элементов



Для связи элементов щелкните на соответствующем контакте (зеленая точка = подходящая точка для связи).

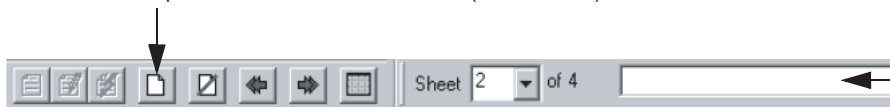
## Создание цикла

Для создания цикла (определения связи в качестве входа) щелкните правой клавишей мыши на связи и выберите *Assume Data Available*.



## Добавление листа

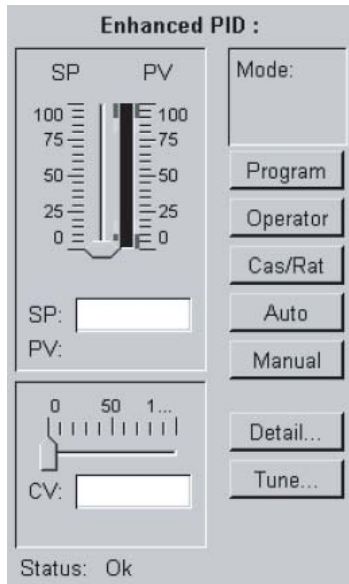
1. Щелкните на кнопке New Sheet (Новый лист).



2. Введите с клавиатуры имя листа.

## Использование разметочной панели для функционального блока

Программное обеспечение RSLogix 5000 имеет разметочные панели для некоторых инструкций функциональных блоков.



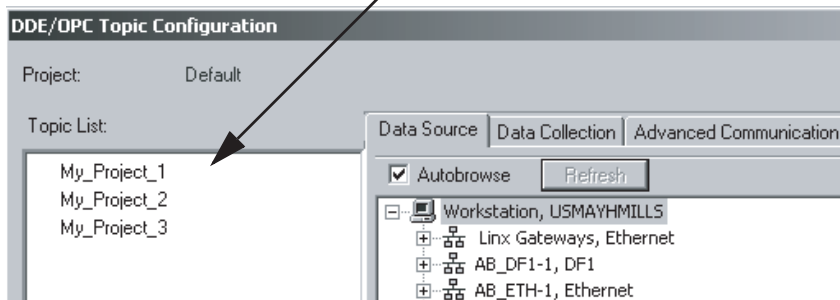
← **Разметочная панель** - средство управления Active-X, позволяющее взаимодействовать с инструкцией языка функциональных блоков.

- Ваш пакет программного обеспечения RSLogix 5000 Enterprise Series включает такое средство как разметочная панель, однако *не устанавливает* его автоматически. Для использования разметочных панелей необходимо найти их на вашем CD и установить отдельно.
- Используйте разметочные панели в контейнере Active-X, такого программного обеспечения как:
  - RSView@32™
  - RSView@ SE
  - Microsoft® Excel
- RSLogix 5000 *не имеет* контейнера Active-X.
- Разметочные панели связываются с контроллером через разделы DDE/OPC в программном обеспечении RSLinx. Для использования RSLinx для разделов DDE/OPC купите либо:
  - RSLinx как отдельный пакет,
  - версию RSLogix 5000 professional, включающую RSLinx professional.
- Программное обеспечение RSLinx Lite, которое поставляется с другими версиями RSLogix 5000, *не обеспечивает* возможность связи DDE/OPC.

Разметочные панели доступны для следующих инструкций:

- Alarm (ALM)
- Enhanced Select (ESEL)
- Totalizer (TOT)
- Ramp/Soak (RMPS)
- Discrete 2-State Device (D2SD)
- Discrete 3-State Device (D3SD)
- Enhanced PID (PIDE)

← **раздел** - в программном обеспечении RSLinx, раздел задает путь к контроллеру.



RSLogix 5000 версии 10.0 или выше автоматически создает раздел RSLinx когда вы:

- создаете проект
- сохраняете проект
- меняете версию проекта на 10.0 или выше.

В некоторых случаях вам необходимо обновить источник данных для раздела в RSLinx.

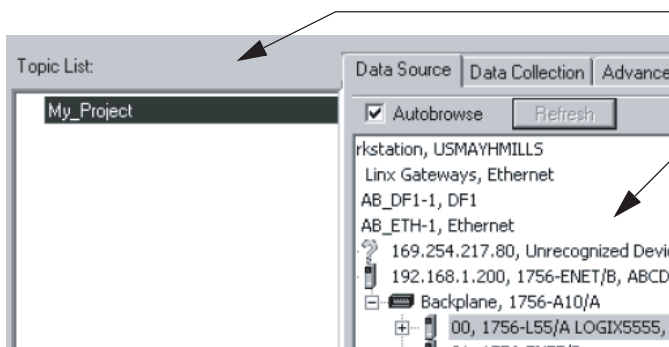
## Создание раздела

1. Для создания раздела используйте программное обеспечение RSLogix 5000:



- a. Настройте путь для проекта (маршрут к контроллеру).
- b. Сохраните проект.

2. В программном обеспечении RSLinx проверьте раздел:

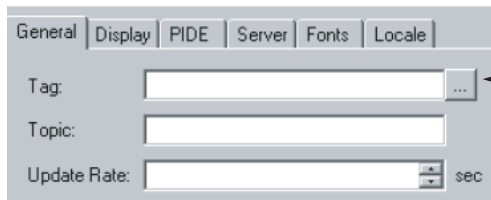


- a. выберите *DDE/OPC => Topic Configuration*.
- b. Выберите проект.
- c. Убедитесь, что источник данных указывает на ваш контроллер.
- d. Выберите

## Добавление разметочной панели к Microsoft Excel



1. Запустите Microsoft Excel.
2. Выберите *View => Toolbars => Control Toolbox*.
3. Щелкните и выберите ту разметочную панель, *Logix 5000...Faceplate Control*, которая вам нужна.
4. В месте расположения разметочной панели отбуксируйте указатель до получения требуемого размера этой разметочной панели.
5. Щелкните правой клавишей мыши на разметочной панели и выберите *Logix 5000...Faceplate Control Object => Properties*.



6. Щелкните и просмотрите теги, которые управляются этой разметочной панелью.
7. Выберите период обновления для управления.
8. Выберите ОК



9. Для выхода из режима настройки и использования этого средства управления щелкните здесь.

## Ввод структурированного текста

Структурированный текст – это текстовый язык программирования, в котором для определения того, что надо выполнить используются операторы. Структурированный текст содержит следующие компоненты:

**конструкция** - определяет логические условия для выполнения других кодов структурированного текста (например, других операторов). В данном примере конструкциями являются If...Then...Else...End\_if.

**логическое выражение (BOOL expression)** - проверяет, является тег или выражение "истиной" или "ложью". Логическое выражение обычно служит в качестве условия для какого-либо действия (if, While или until конструкции).

**присваивание** - записывает значение в тег. Значение перемещается справа от знака := налево.

**численное выражение** - рассчитывает значение.

**точка с запятой ";"** - завершает присваивание, инструкцию или конструкцию.

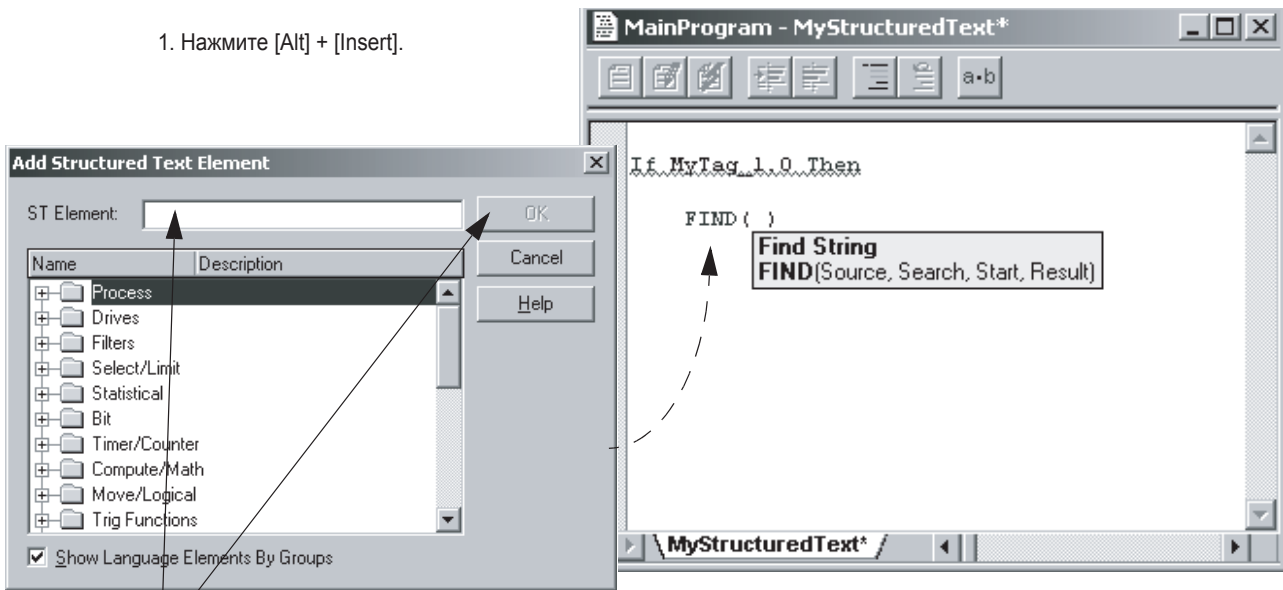
Если вы вводите структурированный текст, то следуйте следующим правилам:

Правило:	Описание:										
1 Структурированный текст не чувствителен к регистру.	Использование любых комбинаций знаков верхнего и нижнего регистров облегчает чтение текста. Например, три способа записи IF, If, if идентичны.										
2 Использование символов табуляции, пробела и возврата каретки (отдельные строки) делает программу легкой для чтения.	Символы табуляции, пробела и возврата каретки не влияют на выполнение структурированного текста. <b>Это:</b> If Bool1 then Bool2 := 1; End_if; <b>Выполняется также как:</b> If Bool1 then Bool2 := 1; End_if; Bool2 := 1; Bool2:=1;										
3 Записывайте логическое выражение либо как «истина» либо как «ложь»	Используйте логическое выражение для определения условий как «истина» (1) или «ложь» (0). • Логический тег (BOOL tag) уже имеет значение «истина» (1) или «ложь» (0). Не используйте символ "=" для проверки этого состояния. <table border="1"> <thead> <tr> <th>правильно</th> <th>неправильно</th> </tr> </thead> <tbody> <tr> <td>If Bool1 ...</td> <td>If Bool1 = 1 ...</td> </tr> <tr> <td>If Not (Bool2) ...</td> <td>If Bool2 = 0 ...</td> </tr> </tbody> </table> • Для проверки целых значений, вещественных или строковых проводите сравнения (=, <, <=, >, >=, <>). <table border="1"> <thead> <tr> <th>правильно</th> <th>неправильно</th> </tr> </thead> <tbody> <tr> <td>If Dint1 &gt; 5 ...</td> <td>If Dint1 ...</td> </tr> </tbody> </table>	правильно	неправильно	If Bool1 ...	If Bool1 = 1 ...	If Not (Bool2) ...	If Bool2 = 0 ...	правильно	неправильно	If Dint1 > 5 ...	If Dint1 ...
правильно	неправильно										
If Bool1 ...	If Bool1 = 1 ...										
If Not (Bool2) ...	If Bool2 = 0 ...										
правильно	неправильно										
If Dint1 > 5 ...	If Dint1 ...										
4 При присваивании начинайте с получателя	Записывайте присваивание в следующем виде: Получатель := Источник ← Данные										



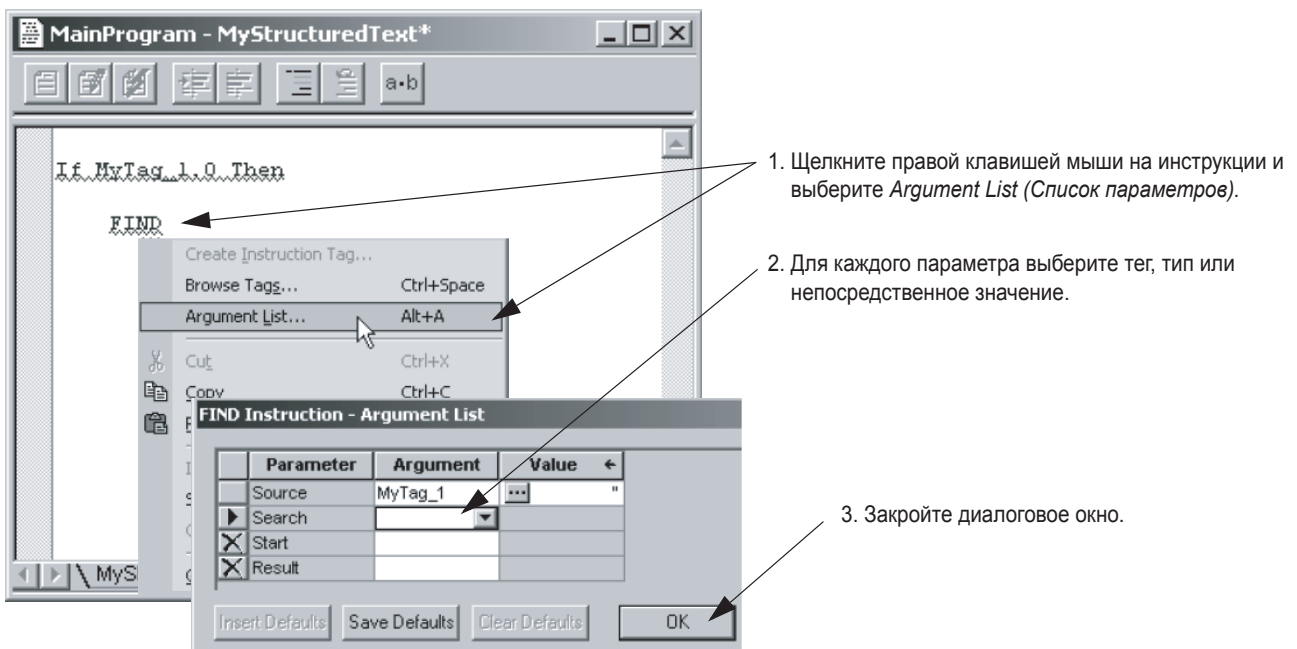
## Просмотр инструкции

1. Нажмите [Alt] + [Insert].



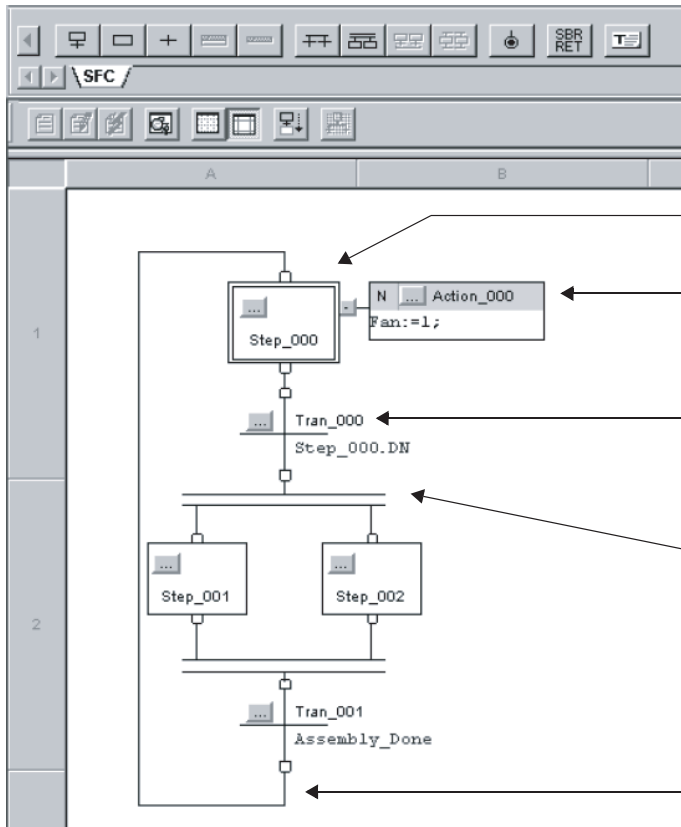
2. Введите с клавиатуры мнемонику для данной инструкции и нажмите [Enter].

## Присваивание операндов инструкции



## Ввод последовательной функциональной схемы

Последовательная функциональная схема (ПФС) позволяет вам определить последовательность шагов, с помощью которых вы управляете процессом. Шаги могут выполнять структурированный текст, вызывать подпрограммы или служить в качестве сигналов для других алгоритмов.



**шаг** - основная функция процесса. Он содержит события, происходящие в определенное время, на определенной фазе или станции.

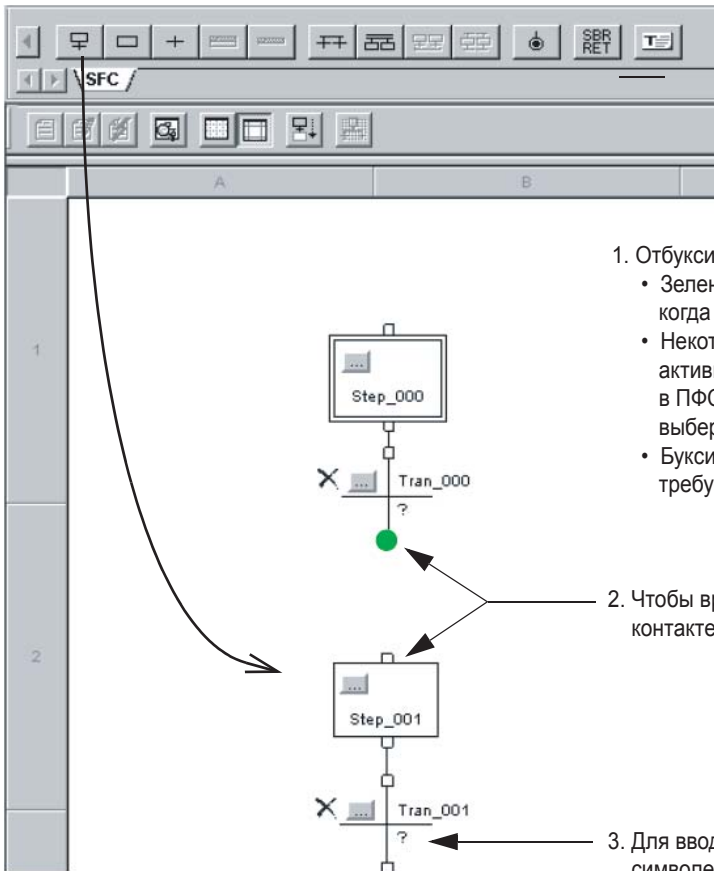
**действие** - одна из функций, выполняемых шагом. Для программирования действия либо введите структурированный текст, либо вызовите подпрограмму.

**переход** - условие "истина" или "ложь", которое сообщает ПФС когда перейти к следующему шагу. Для задания этого условия введите либо логическое выражение на языке структурированного текста, либо вызовите подпрограмму.

**ветвь** - выполнение более одного шага одновременно или выбор между разными шагами.

**связь** - соединение одного элемента с другим в схеме.

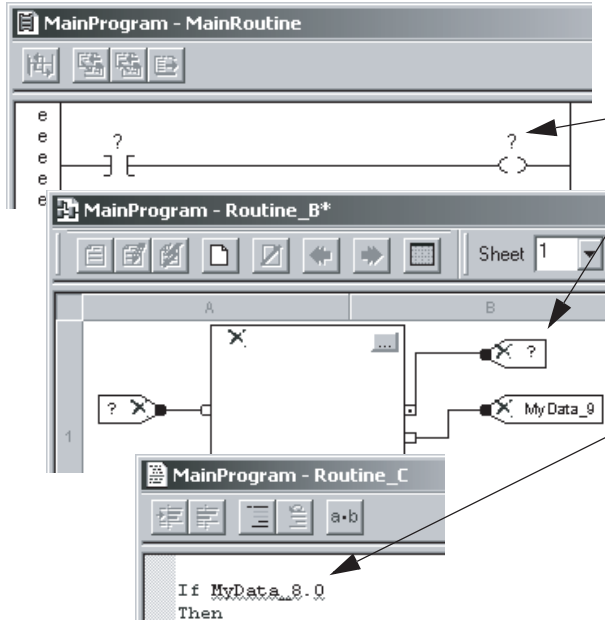
## Ввод ПФС



1. Отбуксируйте элементы из панели инструментов на схему.
  - Зеленая точка указывает, куда будет подключен элемент, когда будет отпущена клавиша мыши.
  - Некоторые кнопки на панели инструментов становятся активными только после выбора соответствующего элемента в ПФС. Например, чтобы добавить действие, вначале выберите шаг.
  - Буксируйте действие, пока оно не окажется наверху требуемого шага, и отпустите клавишу мыши.
2. Чтобы вручную связать элементы, щелкните на соответствующем контакте. Зеленая точка указывает нужное место.
3. Для ввода структурированного текста дважды щелкните на символе "?". Затем введите с клавиатуры структурированный текст и нажмите [Ctrl] + [Enter].

## Присваивание операндов

Программное обеспечение RSLogix 5000 позволяет вам вести программирование в соответствии с вашим рабочим процессом. Вы можете ввести логику без присваивания операндов или определения тегов. Позднее, вы можете вернуться и присвоить операнды и определить теги для завершения процесса программирования.



**пропущенный операнд** - ввод логики без определения операндов. RSLogix 5000 позволяет вводить и сохранять логику без присваивания операндов. Это позволяет разрабатывать алгоритм путем итераций и сохранять библиотеки кодов для повторного использования.

**неопределенный тег** - ввод имени тега без определения этого тега. RSLogix 5000 позволяет вводить и сохранять логику без определения всех операндов. Это позволяет разрабатывать алгоритм путем итераций.

Имя тега имеет формат:

`Name` `[Element]` `.Member` `[Element]` `.Bit`

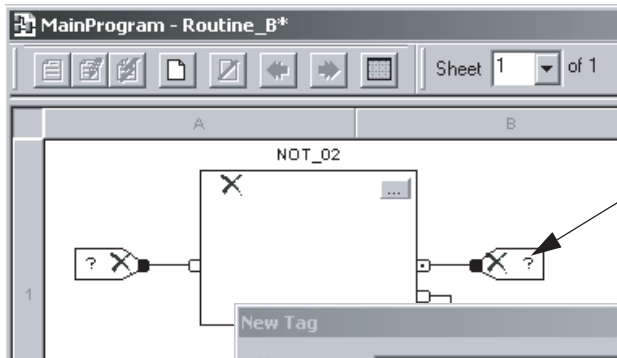
или

`. [Index]`

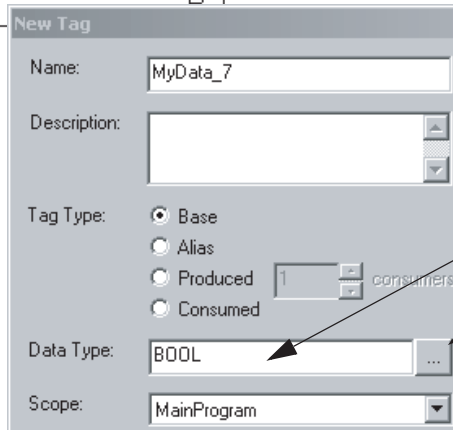
= Необязательны

Где:	Это:
Name	Имя, которое идентифицирует данный тег.
Element	<p>Индекс или индексы, которые указывают на заданный элемент в массиве.</p> <ul style="list-style-type: none"> <li>Используйте идентификатор элемента, только если тег или член являются массивом.</li> <li>Используйте индекс для каждой размерности массива. Например [5], [2,8], [3,2,7].</li> </ul> <p>Для не прямой ссылки на элемент (динамической) используйте тег или числовое выражение, которое предоставляет номер элемента. Например, <code>MyArray[Tag_1]</code>, <code>MyArray[Tag_2-1]</code>, <code>MyArray[ABS(Tag_3)]</code>.</p>
Member	<p>Конкретный член структуры.</p> <ul style="list-style-type: none"> <li>Используйте идентификатор члена, только если данный тег является структурой.</li> <li>Если структура содержит другую структуру в качестве одного из ее членов, то для идентификации требуемого члена используйте дополнительные уровни формата <code>.Member</code></li> </ul>
Bit	Конкретный разряд целочисленного типа данных (SINT, INT или DINT).
Index	Для косвенной (динамической) ссылки на бит целого числа используйте тег или числовое выражение, дающее номер бита. Например, <code>MyTag.[Tag_1]</code> , <code>MyTag.[Tag_2-1]</code> , <code>MyTag.[ABS(Tag_4)]</code> .

## Создание тега

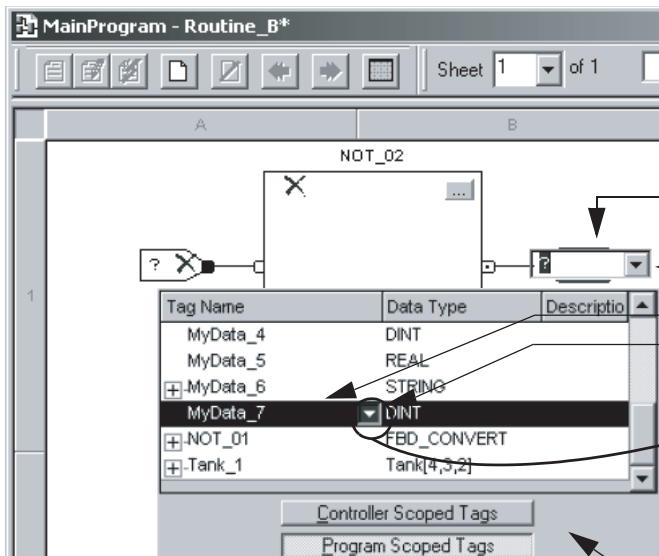


1. Дважды щелкните в области тега.
2. Введите с клавиатуры имя тега и нажмите [Enter]. Вместо пробелов используйте подчеркивание "\_".
3. Щелкните правой клавишей мыши на имени тега и выберите New "Tag\_Name".



4. Введите тип данных.
- Для просмотра типа данных или присваивания размерностей массива щелкните на
5. Выберите для тега область видимости.
6. Выберите

## Выбор существующего тега



1. Дважды щелкните в области тега.

2. Щелкните на

3. Выберите нужный тег.

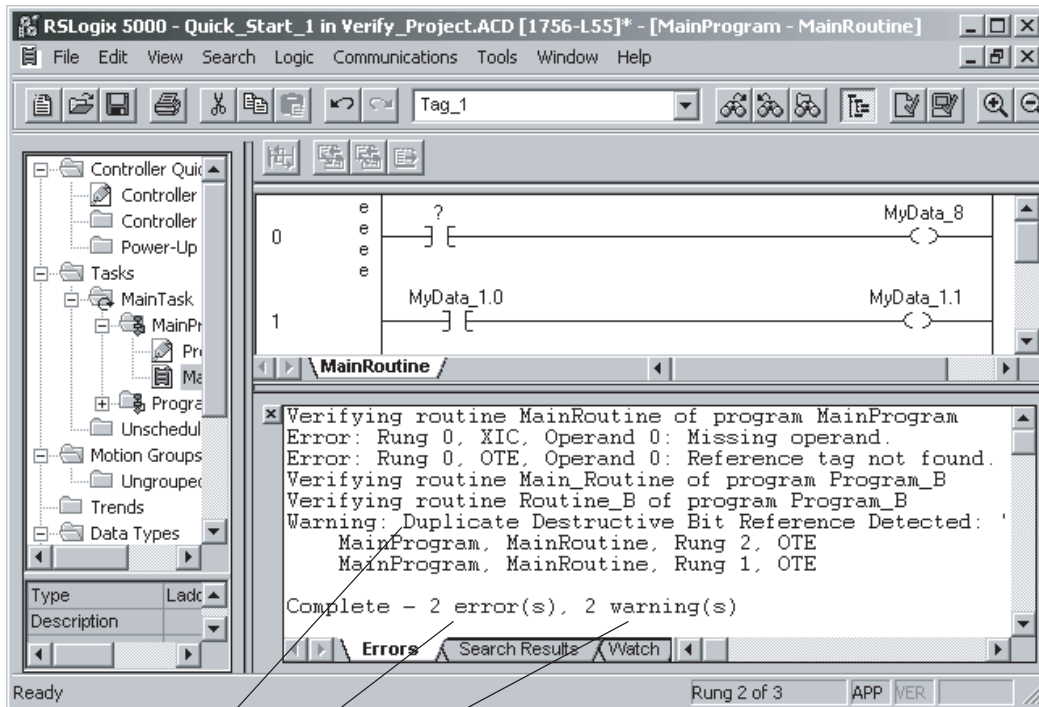
- Чтобы выбрать количество разрядов щелкните на

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

4. Чтобы изменить область видимости тегов, щелкните на соответствующей кнопке.

## Проверка проекта

В процессе программирования периодически проверяйте проект:



**проверка** - проверка процедуры или проекта на предмет наличия программных ошибок или незавершенной конфигурации.

**предупреждение** - ситуация, которая может предотвратить выполнение проекта в том виде, как это ожидалось. Программное обеспечение RSLogix 5000 позволяет загружать проект, включающий предупреждения. Предупреждения включают такие ситуации, как дублирование прерывающих битов и отсутствие главной процедуры.

**ошибка** - ситуацию, которую необходимо исправить перед загрузкой проекта. Ошибки включают такие ситуации как пропуск операнда или неопределенные теги.

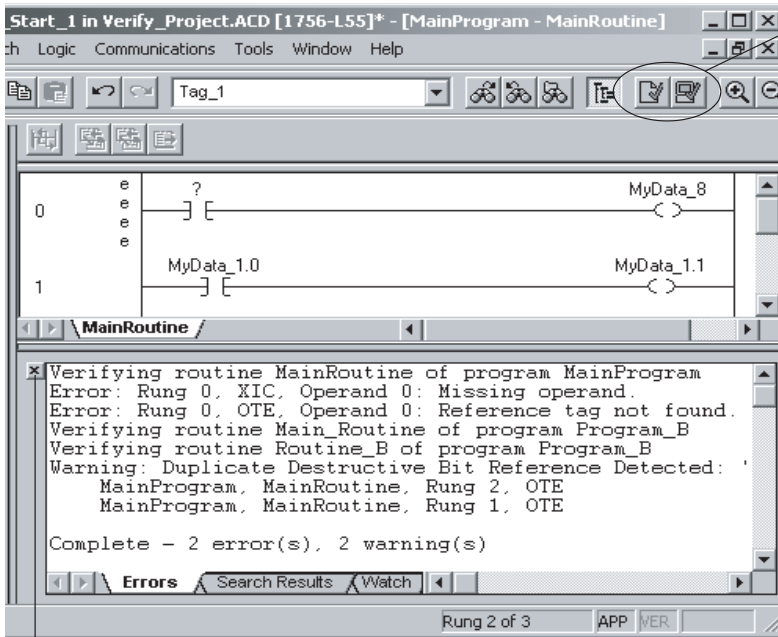
**определение дублирования прерывающих тегов** - определение факта сброса или установки другой логикой (битовая инструкция, OREF, присваивание ST) бита, который вы используете в инструкциях OTE, ONS, OSF или OSR. RSLogix 5000 определяет дублирование прерывающего бита только если имеют место *все* перечисленные ниже условия:

1. Вы разрешаете определение дублирования прерывающего бита (по умолчанию отключено).
2. Вы используете этот бит в инструкциях релейной логики OTE, ONS, OSF или OSR.
3. Другой элемент логики, такой как битовая инструкция OREF или присваивание ST ссылается на тот же самый бит и изменяет его значение.

Если вы *не используете* бит в инструкции OTE, ONS, OSF или OSR, программное обеспечение *не определит* дублирования, даже если оно существует.

По умолчанию определение дублирования прерывающего бита отключено.

Для проверки процедуры или проекта:



1. Выберите опцию проверки:

Проверить данную процедуру

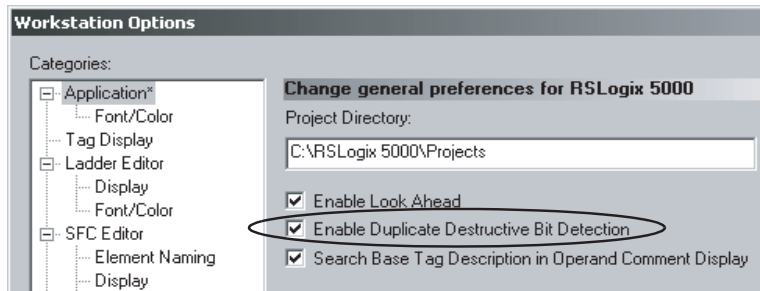
Проверить весь проект

2. Переход к ошибке или предупреждению:

Чтобы перейти к:	Сделайте следующее:
Конкретной ошибке или предупреждению	Дважды щелкните на данной ошибке или предупреждении
Пройти по списку ошибок и предупреждений	Нажмите [F4].

3. Чтобы закрыть закладку *Errors* (*Ошибки*), щелкните здесь.

4. Для того, чтобы разрешить определение дублирования прерывающих битов (не разрешено по умолчанию), выберите *Tools* (*Инструментарий*) => *Options* (*Опции*).



## Руководящие указания по созданию тегов

При создании тегов для проекта Logix5000 придерживайтесь нижеприведенных руководящих указаний:

Указание:	Подробности								
<input type="checkbox"/> 1 Создавайте пользовательские типы данных.	<p>Пользовательские типы данных (структуры) позволяют вам организовать ваши данные таким образом, чтобы они соответствовали вашей установке или процессу.</p> <p>Пользовательский тип данных имеет следующие преимущества:</p> <ul style="list-style-type: none"> <li>• Один тег содержит все данные по определенному аспекту вашей системы. Это обеспечивает совместное размещение взаимосвязанных данных и простоту их поиска независимо от типа данных.</li> <li>• Каждая отдельная часть данных (член) получает описательное имя. Это автоматически создает начальный уровень документирования вашей логики.</li> <li>• Вы можете использовать этот тип данных для создания множества тегов при том же макете данных.</li> </ul> <p>Например, используйте пользовательский тип данных для хранения всех параметров резервуара, включая температуры, давления, позиции клапанов и предварительно установленные значения. Затем создайте тег для каждого из ваших резервуаров на основе этого типа данных.</p>								
<input type="checkbox"/> 2 Используйте массив для быстрого создания группы аналогичных тегов.	<p>Массив создает множество экземпляров данных определенного типа под общим именем тега.</p> <ul style="list-style-type: none"> <li>• Массивы позволяют вам организовать блок тегов, использующих один и тот же тип данных и выполняющих аналогичную функцию.</li> <li>• Организуйте данные в одно-, двух- или трехмерные массивы в зависимости от того, что отображают эти данные.</li> </ul> <p>Например, используйте двухмерный массив для организации данных по резервуарному парку. Каждый элемент массива соответствует одному резервуару. Местоположение элемента в массиве соответствует географическому расположению резервуара.</p> <p><b>Важно:</b> Минимизируйте использование массивов BOOL. Многие инструкции массивов <i>не работают</i> для массивов BOOL. Это затрудняет инициализацию и очистку массива данных типа BOOL.</p> <ul style="list-style-type: none"> <li>• Используйте массив BOOL в основном для объектов битового уровня экрана PanelView.</li> <li>• Также можно использовать отдельные биты тега DINT или массив данных типа DINT.</li> </ul>								
<input type="checkbox"/> 3 Воспользуйтесь тегами в области видимости программы.	<p>Если вы хотите, чтобы у вас было множество тегов с одним и тем же именем, задайте каждый тег как программный тег (тег в области видимости программы) для другой программы. Это позволяет вам многократно использовать в разных программах и логику, и имена тегов.</p> <p>Избегайте использования одного и того же имени для тегов контроллера и программных тегов. Внутри программы вы не можете обращаться к тегу контроллера, если существует программный тег с таким же именем для этой программы.</p> <p>Определенные теги должны иметь область видимости контроллера (тег контроллера).</p> <table border="1"> <thead> <tr> <th>Если вы хотите использовать тег:</th> <th>Задайте эту область видимости:</th> </tr> </thead> <tbody> <tr> <td>в нескольких программах проекта</td> <td rowspan="3">область видимости контроллера (controller scope) (теги контроллера)</td> </tr> <tr> <td>в инструкции Message (MSG)</td> </tr> <tr> <td>в инструкции Message (MSG) для обмена данными с терминалом PanelView</td> </tr> <tr> <td>для других целей, кроме вышеперечисленного</td> <td>область видимости программы (program scope) (теги программы)</td> </tr> </tbody> </table>	Если вы хотите использовать тег:	Задайте эту область видимости:	в нескольких программах проекта	область видимости контроллера (controller scope) (теги контроллера)	в инструкции Message (MSG)	в инструкции Message (MSG) для обмена данными с терминалом PanelView	для других целей, кроме вышеперечисленного	область видимости программы (program scope) (теги программы)
Если вы хотите использовать тег:	Задайте эту область видимости:								
в нескольких программах проекта	область видимости контроллера (controller scope) (теги контроллера)								
в инструкции Message (MSG)									
в инструкции Message (MSG) для обмена данными с терминалом PanelView									
для других целей, кроме вышеперечисленного	область видимости программы (program scope) (теги программы)								



Указание:	Подробности										
<p><input type="checkbox"/> 4 Для целочисленных значений используйте тип данных DINT.</p>	<p>Для увеличения эффективности вашей логики минимизируйте использование типов данных SINT и INT. Везде, где это возможно, для целочисленных значений используйте тип данных DINT.</p> <ul style="list-style-type: none"> <li>• контроллер Logix5000 обычно выполняет сравнение или манипуляции со значениями как с 32-разрядными величинами (тип данных DINT или REAL).</li> <li>• как правило, перед использованием значения контроллер преобразует значение SINT или INT в DINT или REAL.</li> <li>• если приемником данных является тег SINT или INT, то контроллер обычно преобразует значение обратно в SINT или INT.</li> <li>• преобразование в или из SINT или INT происходит автоматически, не требуя никакого дополнительного программирования. Однако, оно требует дополнительного времени выполнения и дополнительной памяти.</li> </ul>										
<p><input type="checkbox"/> 5 Ограничьте имя тега 40 символами.</p>	<p>Правила для имени тега:</p> <ul style="list-style-type: none"> <li>• допускаются только буквы (A-Z или a-z), цифры (0-9) и знаки подчеркивания (_)</li> <li>• должно начинаться с буквы или знака подчеркивания</li> <li>• может включать не более 40 символов</li> <li>• не должно включать несколько знаков подчеркивания подряд или замыкающий знак подчеркивания (_)</li> <li>• не чувствительно к регистру</li> </ul>										
<p><input type="checkbox"/> 6 Используйте символы разных регистров</p>	<p>Хотя имена тегов не чувствительны к регистру (буква A верхнего регистра воспринимается так же, как буква a нижнего регистра), имена легче читаются при использовании символов разных регистров.</p> <table border="1" data-bbox="467 1064 1445 1310"> <thead> <tr> <th data-bbox="467 1064 1005 1108">Эти теги легче читаются:</th> <th data-bbox="1013 1064 1445 1108">Чем эти теги:</th> </tr> </thead> <tbody> <tr> <td data-bbox="467 1108 1005 1153">Tank_1</td> <td data-bbox="1013 1108 1445 1153">TANK_1</td> </tr> <tr> <td data-bbox="467 1153 1005 1198">Tank1</td> <td data-bbox="1013 1153 1445 1198">TANK1</td> </tr> <tr> <td data-bbox="467 1198 1005 1243"></td> <td data-bbox="1013 1198 1445 1243">tank_1</td> </tr> <tr> <td data-bbox="467 1243 1005 1288"></td> <td data-bbox="1013 1243 1445 1288">tank1</td> </tr> </tbody> </table>	Эти теги легче читаются:	Чем эти теги:	Tank_1	TANK_1	Tank1	TANK1		tank_1		tank1
Эти теги легче читаются:	Чем эти теги:										
Tank_1	TANK_1										
Tank1	TANK1										
	tank_1										
	tank1										
<p><input type="checkbox"/> 7 Учитывайте алфавитный порядок тегов.</p>	<p>Программное обеспечение RSLogix5000 показывает на экране теги одной и той же области видимости в алфавитном порядке. Для облегчения контроля взаимосвязанных тегов используйте одинаковые начальные символы для тегов, которые должны отображаться вместе.</p> <div style="display: flex; justify-content: space-around;"> <div data-bbox="467 1456 845 1758"> <p><b>Если каждый тег для резервуара будет начинаться с <input type="checkbox"/> <input type="checkbox"/>, то эти теги будут отображаться вместе.</b></p> <table border="1" data-bbox="478 1568 686 1758"> <thead> <tr> <th>Tag Name</th> </tr> </thead> <tbody> <tr> <td>Tank_North</td> </tr> <tr> <td>Tank_South</td> </tr> <tr> <td>...</td> </tr> </tbody> </table> </div> <div data-bbox="925 1456 1445 1848"> <p><b>В противном случае эти теги могут оказаться далеко друг от друга.</b></p> <table border="1" data-bbox="925 1568 1133 1848"> <thead> <tr> <th>Tag Name</th> </tr> </thead> <tbody> <tr> <td>North_Tank</td> </tr> <tr> <td>...</td> </tr> <tr> <td>...</td> </tr> <tr> <td>...</td> </tr> <tr> <td>South_Tank</td> </tr> </tbody> </table> <p>← другие теги, начинающиеся с букв o, p, q и т.д.</p> </div> </div>	Tag Name	Tank_North	Tank_South	...	Tag Name	North_Tank	...	...	...	South_Tank
Tag Name											
Tank_North											
Tank_South											
...											
Tag Name											
North_Tank											
...											
...											
...											
South_Tank											

**Для заметок:**

---

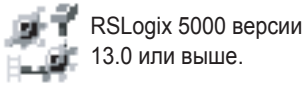
## Документирование проекта

### Использование данной главы

Используйте данную главу при документировании проекта RSLogix 5000. Это облегчит отладку, обслуживание и диагностику ошибок.

Если вы хотите:	См. стр.
Описать пользовательский тип данных	4-2
Добавить к цепочке комментарии	4-4
Ввести и отредактировать комментарии, используя Microsoft® Excel	4-5
Добавить комментарии к функциональной блок-схеме или ПФС	4-7
Добавить комментарии к структурированному тексту	4-9

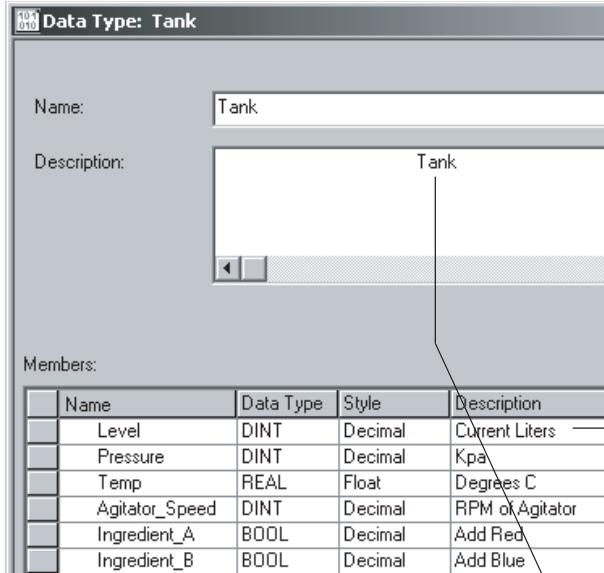
## Описание пользовательского типа данных



RSLogix 5000 версии 13.0 или выше.

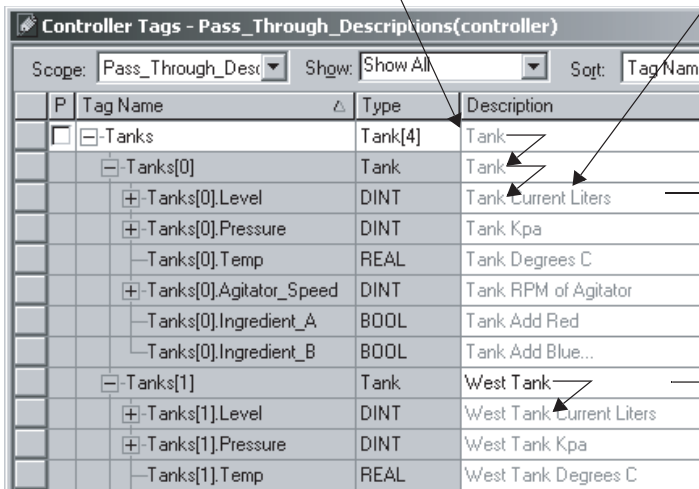
Программное обеспечение RSLogix 5000 позволяет вам автоматически создавать описания на основе описаний, содержащихся в ваших пользовательских типах данных. Это значительно сокращает время, затрачиваемое вами на документирование вашего проекта.

При организации пользовательских типов данных помните о следующих возможностях программного обеспечения RSLogix 5000:



**сквозное использование описаний** - Когда это возможно, RSLogix 5000 ищет имеющееся описание для тега, элемента или члена:

- Описания в пользовательских типах данных переносятся на теги, использующие соответствующий тип данных.
- Описание тега массива переносится на элементы и члены этого массива.



**присоединение описания к базовому тегу** - RSLogix 5000 автоматически формирует описание для каждого члена тега, использующего пользовательский тип данных. Оно начинается с описания данного тега, к которому присоединяется описание члена, взятое из типа данных.

**вставка сквозного описания** - Используйте описание типа данных и массива в качестве основы для более конкретных описаний. В данном примере Tank (Резервуар) превратился в West Tank (Западный резервуар).

Для описаний RSLogix 5000 использует различные цвета:

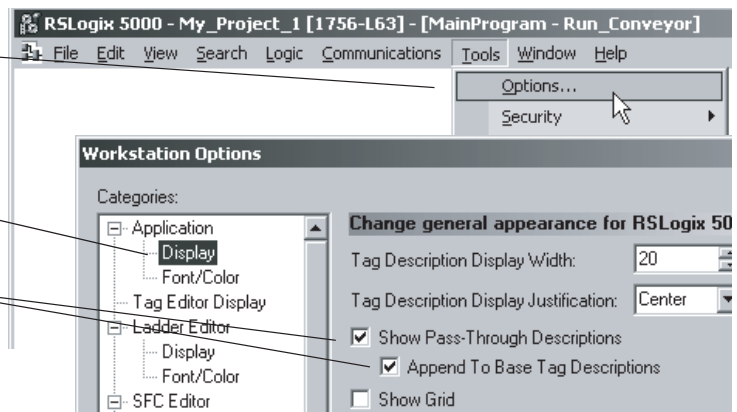
Описание этого цвета:	Соответствует:
серый	сквозному описанию
черный	описанию, введенному вручную

## Включение и выключение сквозных (pass-through) и присоединяемых (append) описаний

1. В RSLogix 5000 выберите *Tools (Сервис) => Options (Опции)*.

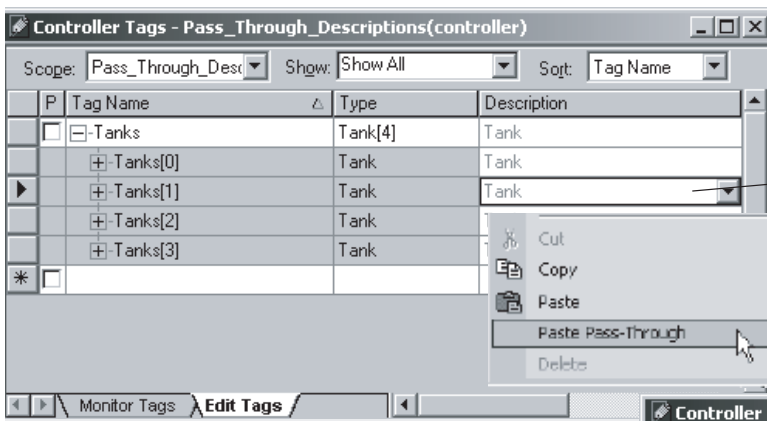
2. Выберите *Application (Приложение) => Display (Отображение)*.

3. Включите (отметьте) или выключите (снимите отметку) соответствующие опции.



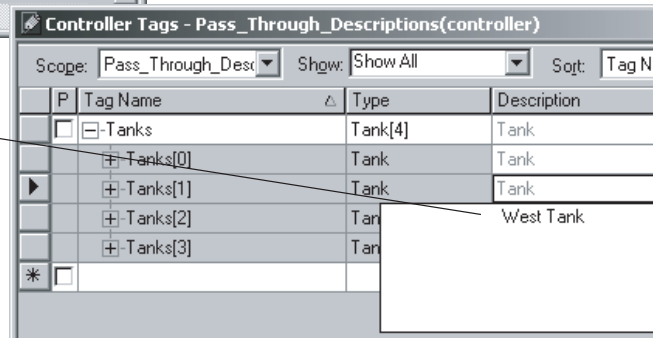
## Вставка сквозных описаний в RSLogix

Для использования сквозного описания в качестве основы для более конкретного описания:



1. Щелкните правой кнопкой мыши по сквозному описанию и выберите *Paste Pass-Through (Вставка сквозного описания)*.

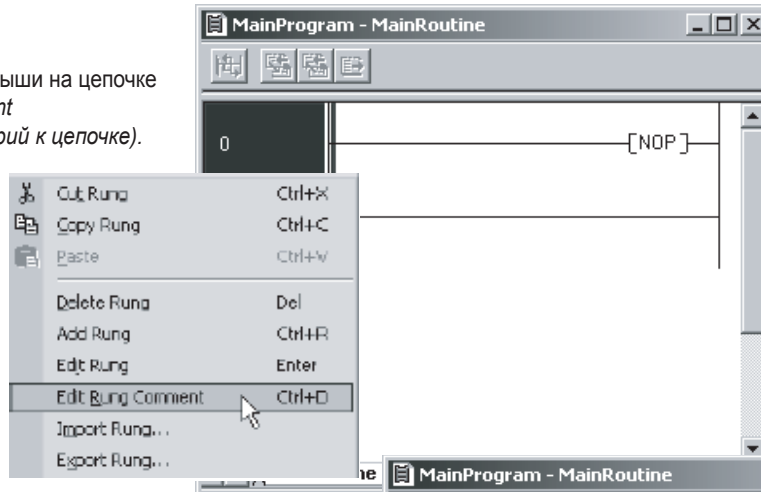
2. Отредактируйте описание и нажмите [Ctrl] + [Enter].



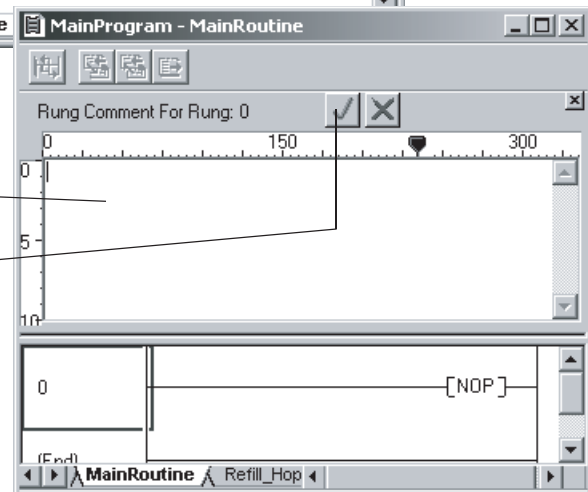
## Добавление комментария к цепочке

Комментарии используются для описания работы цепочки релейной логики. Вы можете запустить процедуру с цепочкой, которая содержит только инструкцию No Operation (NOP) (Нет операций). Добавляйте комментарий к этой начальной цепочке для общего описания процедуры.

1. Щелкните правой клавишей мыши на цепочке и выберите *Edit Rung Comment* (Комментировать комментарий к цепочке).



2. Введите с клавиатуры ваш комментарий.



3. Закройте окно ввода.

## Ввод и редактирование комментария с использованием Microsoft® Excel



RSLogix 5000 версии 13.0 или выше.

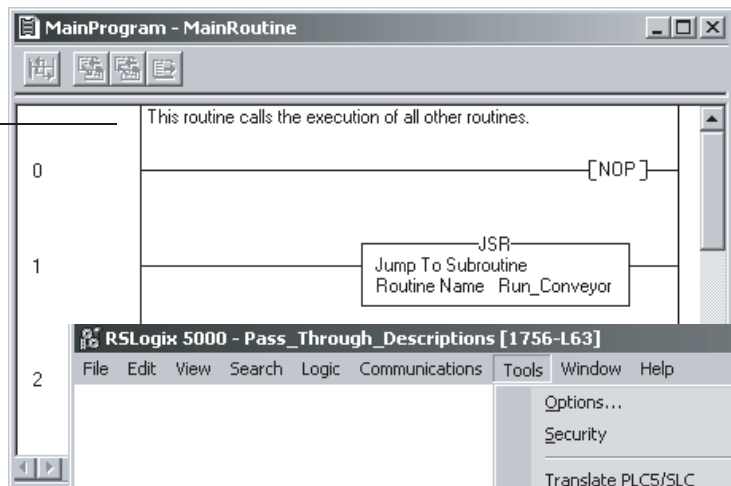
Для создания и редактирования комментариев к цепочке, вы можете использовать программное обеспечение, работающее с электронными таблицами, такое как   . Это дает вам все преимущества, которое дает программное обеспечение, работающее с электронными таблицами.

### ВАЖНО

Комментарий к цепочке экспортируется в формате CSV (comma delimited). При сохранении или экспорте файла убедитесь, что этот формат поддерживается.

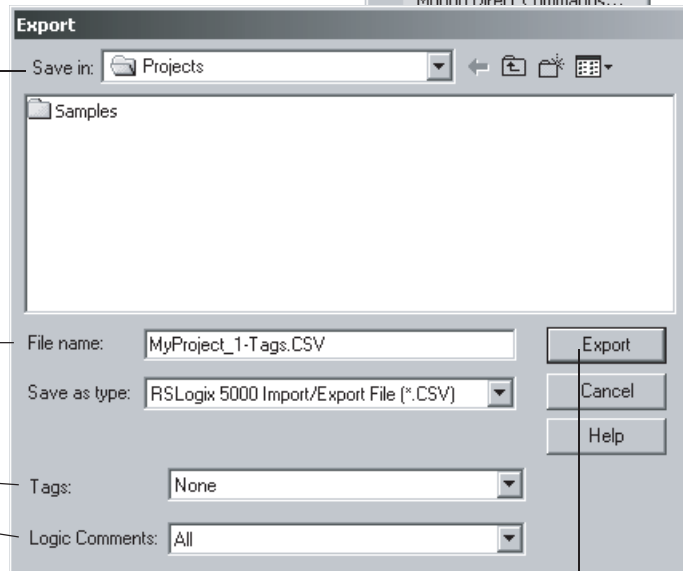
### Экспорт существующего комментиря

1. В RSLogix 5000 добавьте хотя бы одну цепочку с комментарием. Это поможет отформатировать экспортируемый файл



2. Выберите *Tools => Export*.

3. Определите расположение и имя экспортируемого файла.



4. Выберите что экспортировать.

5. Нажмите Export.

## Редактирование экспортируемого файла

1. Откройте экспортный файл в Microsoft Excel.
2. Введите комментарий к цепочке в следующем формате:

	A	B	C	D	E	F
7	TYPE	SCOPE	ROUTINE	COMMENT	OWNING_ELEMENT	LOCATION
8	RCOMMENT	MainProgram	MainRoutine	This routine calls the execution of all other routines.	NOP()	0
9	RCOMMENT	MainProgram	MainRoutine	If the conveyor is not turning on or off, check this routine.		1
10						

RCOMMENT	программа, содержащая цепочку	процедура, содержащая цепочку	комментарий к цепочке	пропуск	номер цепочки
----------	-------------------------------	-------------------------------	-----------------------	---------	---------------

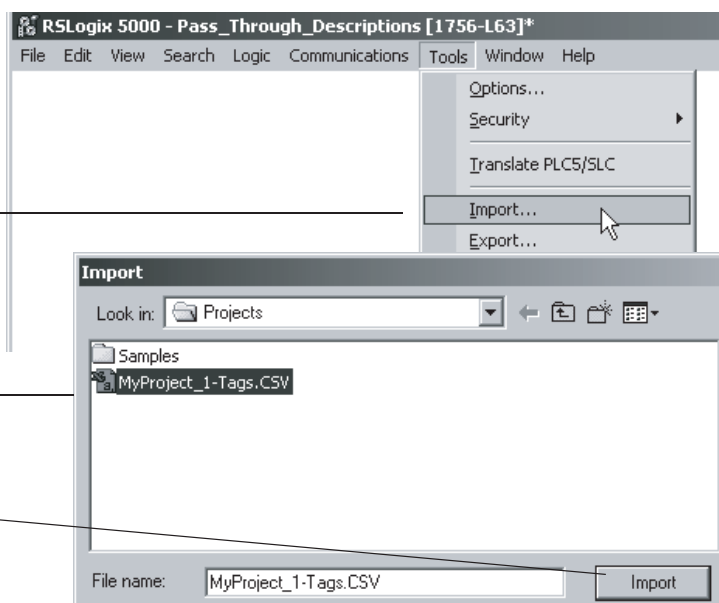
3. Сохраните и закройте файл (храните его в формате CSV).

## Импортирование новых комментариев

1. В RSLogix 5000 выберите *Tools => Import*.

2. Выберите комментарии, которые вы вводите (например, экспортный файл).

3. Import (Импорт).



Проверьте закладку *Errors* на предмет проверки результата операции импортирования. Для обновления отображения релейной логики и просмотра комментария, закройте и откройте процедуру.

```
Totals:
  0 tag(s) created
  0 tag(s) overwritten on collision
  0 description(s) imported
  1 new comment(s) imported
  0 comment(s) overwritten on collision
Complete - 0 error(s), 0 warning(s)
```

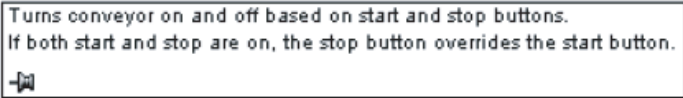
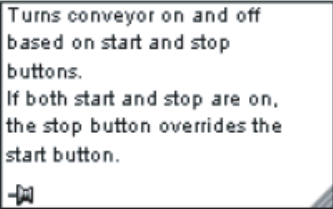


## Добавление комментария к функциональной блок-схеме или ПФС

Для добавления комментария в блок-схеме или вблизи схемы конкретного элемента используйте текстовые окна. Или же используйте текстовое окно для хранения информации, которую вы используете позднее по мере разработки проекта.

### Настройка опции Word Wrap (Укладка текста)

Используйте опцию word wrap для настройки ширины текстового окна при вводе с клавиатуры. Вы настраиваете эту опцию независимо для функциональной блок-схемы и ПФС.

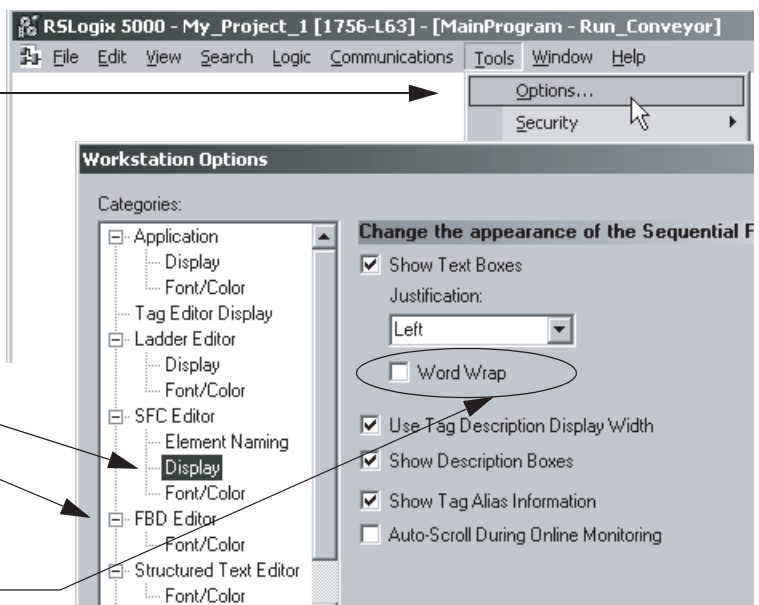
Если вы хотите, чтобы текстовое окно:	Выберите эту опцию:
<p>Автоматически увеличивало ширину по длинной строке текста.</p> 	<input type="checkbox"/> Word Wrap
<p>Сохраняло фиксированную ширину и упаковывало текст. Вы всегда можете вручную изменить размер окна.</p> 	<input checked="" type="checkbox"/> Word Wrap

Для настройки опции Word Wrap:

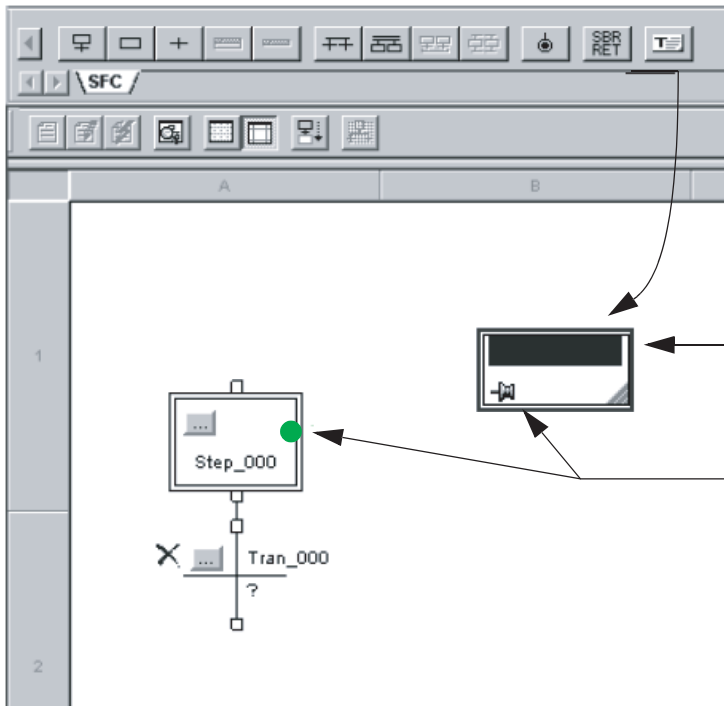
1. В RSLogix 5000 выберите Tools => Options.

2. Выберите редактор.

3. Пометьте или очистите опцию word wrap.



### Добавление текстового окна



1. Отбуксируйте текстовое окно из панели инструментов на схему.
2. Введите с клавиатуры комментарий и нажмите [Ctrl] + [Enter]
3. Для привязки этого текстового окна конкретному элементу, щелкните на символе контакта, а затем на соответствующем элементе. Зеленая точка указывает точку контакта.

## Добавление комментария к структурированному тексту

Используйте комментарии для того, чтобы сделать программу на языке структурированного текста более понятной при чтении. Комментарии:

- позволяют вам использовать простой язык для описания работы программы на языке структурированного текста
- могут быть загружены в контроллер и выгружены из него
- комментарии *не влияют* на выполнение программы

Чтобы добавить комментарии к программе на языке структурированного текста:

Чтобы добавить комментарий:	Используйте следующие форматы:
на одной строке	//comment
в конце строки структурированного текста	(*comment*) /*comment*/
внутри одной строки структурированного текста	(*comment*) /*comment*/
если занимает более одной строки	(*начало комментария ... конец комментария*) /*начало комментария ... конец комментария*/

Например:

Формат:	Пример:
//comment	<p><b>В начале строки</b> //Check conveyor belt direction IF conveyor_direction THEN...</p> <p><b>В конце строки</b> ELSE //If conveyor isn't moving, set alarm light light := 1; END_IF;</p>
(*comment*)	<p>Sugar.Inlet[:]=1;(*open the inlet*)</p> <p>IF Sugar.Low (*low level LS*)&amp; Sugar.High (*high level LS*) THEN...</p> <p>(*Controls the speed of the recirculation pump. The speed depends on the temperature in the tank.*) IF tank.temp &gt; 200 THEN...</p>
/*comment*/	<p>Sugar.Inlet:=0;/*close the inlet*/</p> <p>IF bar_code=65 /*A*/ THEN...</p> <p>/*Gets the number of elements in the Inventory array and stores the value in the Inventory_Items tag*/ SIZE(Inventory,0,Inventory_Items);</p>

**Для заметок:**

## **Работа с контроллером в режиме онлайн**

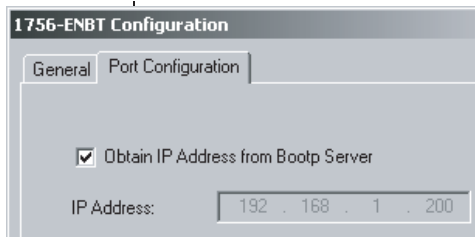
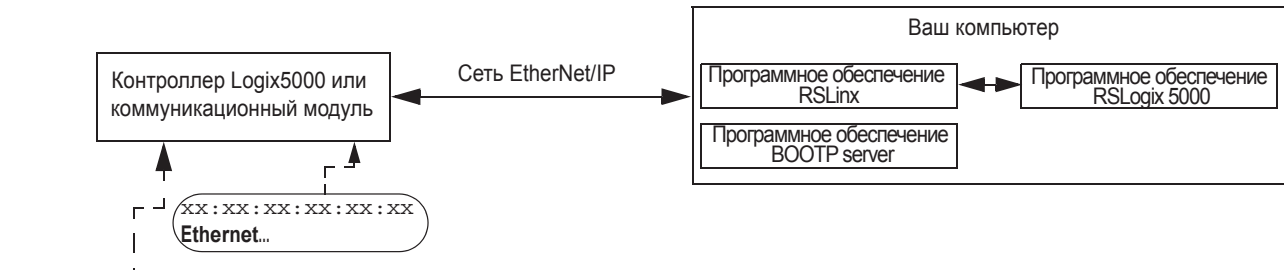
### **Использование данной главы**

Используйте данную главу при получении доступа к проекту в контроллере с целью отслеживания его работы, редактирования или диагностики ошибок.

<b>Для:</b>	<b>См. Стр.</b>
Установления связи с контроллером через EtherNet/IP	5-2
Работы в режиме онлайн	5-6

## Настройка связи с контроллером через EtherNet/IP

RSLink® обеспечивает связь между контроллерами Logix5000 и вашим программным обеспечением, таким как RSLogix 5000. Для связи с контроллером (т.е. загрузки и контроля данных) сконфигурируйте соответствующим образом RSLink.



**Адрес ethernet (MAC)** - адрес, присваиваемый модулю производителем.

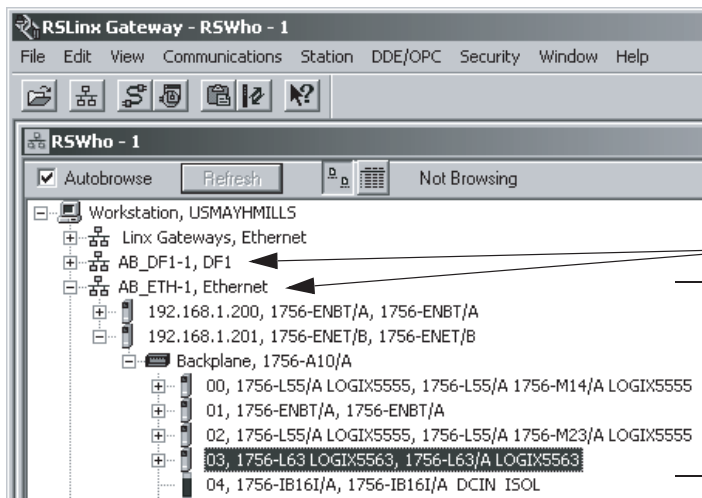
- Модуль всегда хранит свой адрес ethernet.
- Для определения ethernet адреса устройства, взгляните на наклейку на этом устройстве.
- Для ethernet адреса используется формат:  
xx : xx : xx : xx : xx : xx

**Адрес IP** - адрес, который вы присваиваете модулю для связи через заданную сеть ethernet. Для адреса IP используется формат:  
xxx . xxx . xxx . xxx

**BOOTP** - конфигурирует устройство на запрос адреса IP по сети ethernet от BOOTP server. Исходно, устройства Allen-Bradley EtherNet/IP сконфигурированы на протокол BOOTP.

**BOOTP server** - программное обеспечение, получающее запросы BOOTP от устройств Ethernet, и присваивающее адреса IP. Программное обеспечение RSLink версии 2.40 и выше включает в себя BOOTP server.

**Драйвер (driver)** - обеспечивает связь через заданную сеть.



**Путь (path)** - коммуникационный маршрут к устройству. Для задания пути, раскройте драйвер и выберите устройство.

## Нужное вам оборудование и информация

1. В зависимости от типа контроллера, вам может понадобиться коммуникационный модуль или дочерняя плата □

Если у вас контроллер:	Установите:	В этот месте:
1756 ControlLogix	модуль □ □□ □□□	Свободный слот на том же шасси, что и контроллер
1769-L35E CompactLogix	Не требуется дополнительного коммуникационного модуля или платы	
1794 FlexLogix	Дочерняя плата □ □□	Свободный слот в контроллере

2. Для устройства □ □□ (контроллер, модуль моста или дочерняя плата) необходимо следующее:

Получите:	Из:
Адрес □□□	Наклейка на устройстве
Адрес IP	Администратор сети
Маска подсети	
Адрес шлюза (может не понадобиться)	

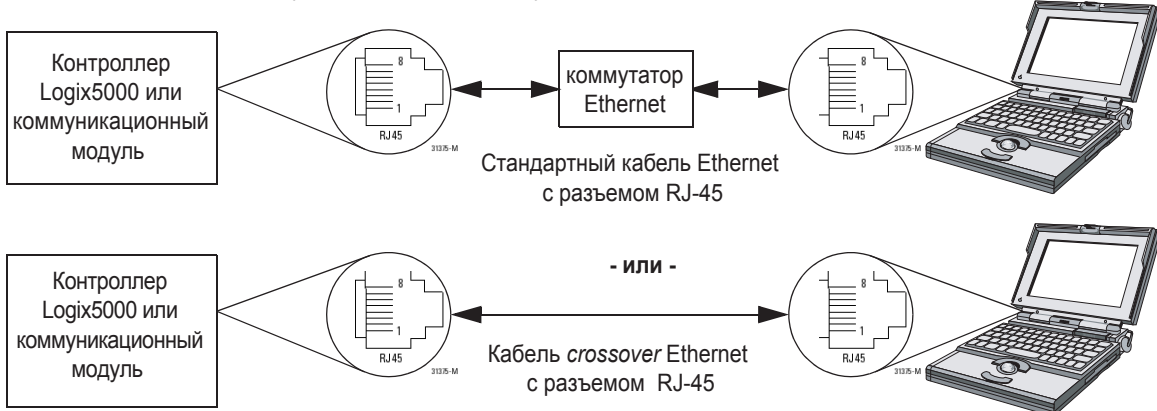
## Подключение устройства EtherNet/IP к компьютеру

### ПРЕДУПРЕЖДЕНИЕ



Если вы подключаете или отключаете коммуникационные кабели при включенном питании устройства или сети, может возникнуть электрическая искра. Это может вызвать взрыв в случае установки в зонах с повышенной опасностью.

Подключите ваше EtherNet/IP устройство к компьютеру через кабель Ethernet.



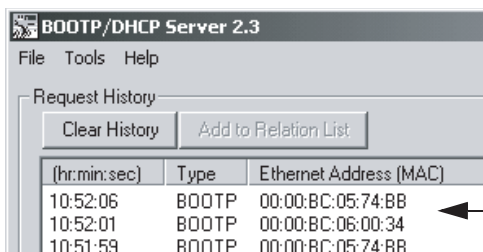
## Присваивание адреса IP контроллеру или коммуникационному модулю

Если у вас *нет* последовательной связи с контроллером

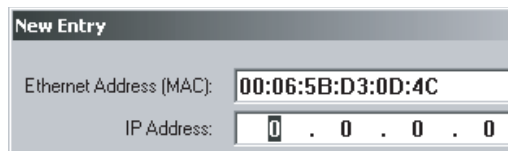
1. Запустите программное обеспечение BOOTP server:  
Start => Programs => Rockwell Software => BOOTP-DHCP Server => BOOTP-DHCP Server  
- или -  
Start => Programs => Rockwell Software => RSLinx Tools => BOOTP-DHCP Server.



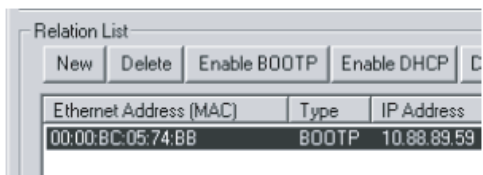
2. Если вы используете программное обеспечение первый раз, введите с клавиатуры маску подсети и шлюз (если нужно) для вашей сети и выберите **OK**.



3. Дважды щелкните на адресе ethernet контроллера/коммуникационного модуля.



4. Введите с клавиатуры адрес IP и выберите **OK**.




5. В Relation List (нижняя часть), выберите устройство и **Disable BOOTP/DHCP**.

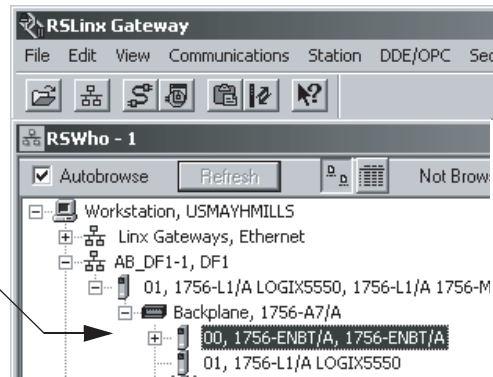
Это позволит вашему устройству сохранить адрес даже после выключения и включения питания.

6. Когда вы выбираете программное обеспечение BOOTP, вам предлагается сохранить изменения.
  - Если вы хотите записать адрес IP, который вы присвоили устройству, сохраните изменения.
  - Независимо от того, сохраняете вы изменения или нет, устройство сохраняет адрес IP.





**Если у вас есть последовательная связь с контроллером**

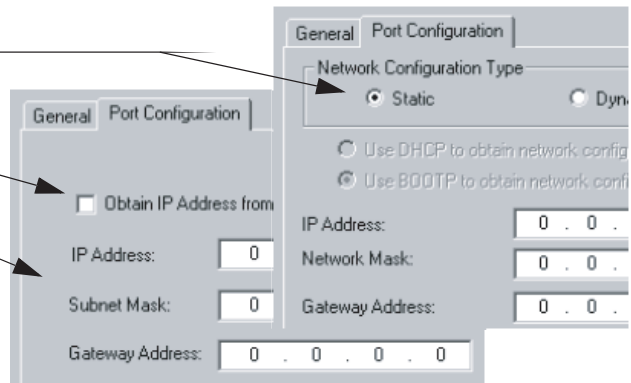
1. Запустите RSLinx.
2. Щелкните 
3. Найдите устройство EtherNet/IP.  
Чтобы открыть уровень, щелкните на значке +.
4. Щелкните правой клавишей мыши на устройстве и выберите *Module Configuration (Конфигурация модуля)*.
5. Щелкните на закладке Port Configuration (Конфигурация порта).



6. В зависимости от устройства либо:
  - Выберите Static (Статический).
  - Снимите флажок *Obtain IP Address from Bootp Server*.


7. Введите с клавиатуры:
  - адрес IP
  - маску подсети
  - адрес шлюза (если требуется).

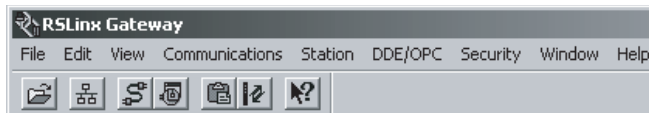
8. Выберите  и затем  (yes-смена адреса IP).



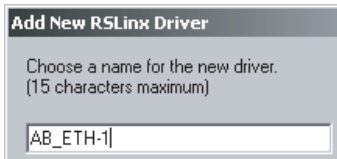
**Конфигурирование драйвера Ethernet**

1. Запустите RSLinx.

2. Щелкните на 

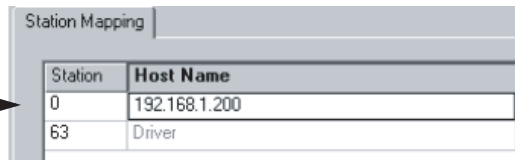


3. Выберите *Ethernet devices (Устройство Ethernet)* и выберите 



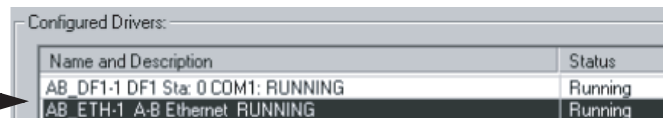
4. Примите имя по умолчанию.

5. Введите с клавиатуры адрес IP контроллера или коммуникационного модуля.



6. Выберите 

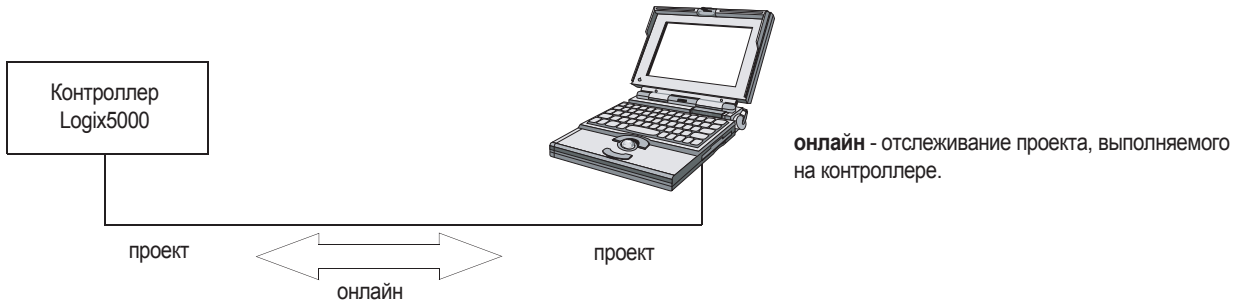
Драйвер успешно сконфигурирован и запущен.



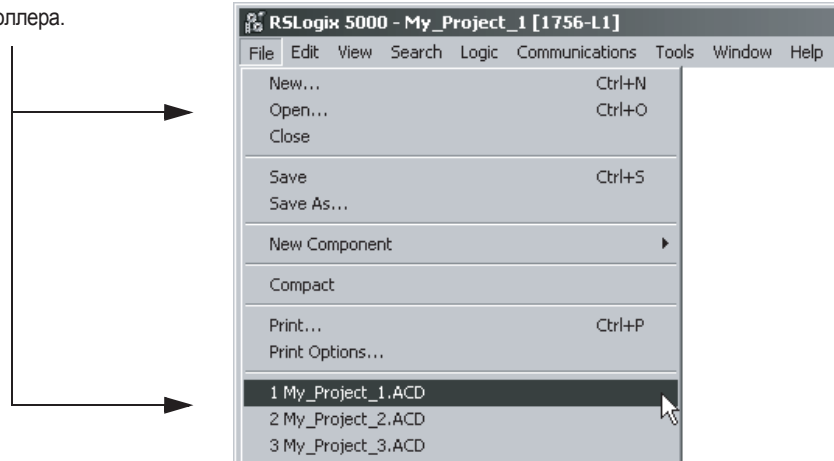
## Работа с контроллером в режиме онлайн

Работа в режиме онлайн с контроллером необходима для отслеживания проекта, выполняемого на контроллере. Процедура, которую вы используете для этого, зависит от того, имеете ли вы копию проекта на компьютере.

### Если на вашем компьютере есть проект, выполняемый на контроллере...



1. Откройте проект RSLogix 5000 для контроллера.



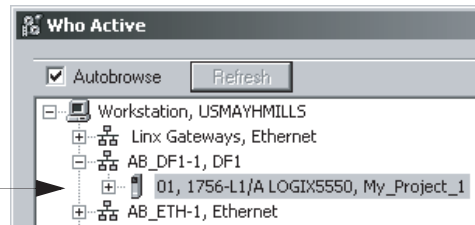
2. Определите путь к контроллеру:



a. Щелкните на

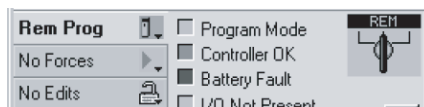
b. Выберите контроллер.

- Чтобы открыть уровень, щелкните на значке +.
- Если контроллер уже выбран, убедитесь что это требуемый контроллер.

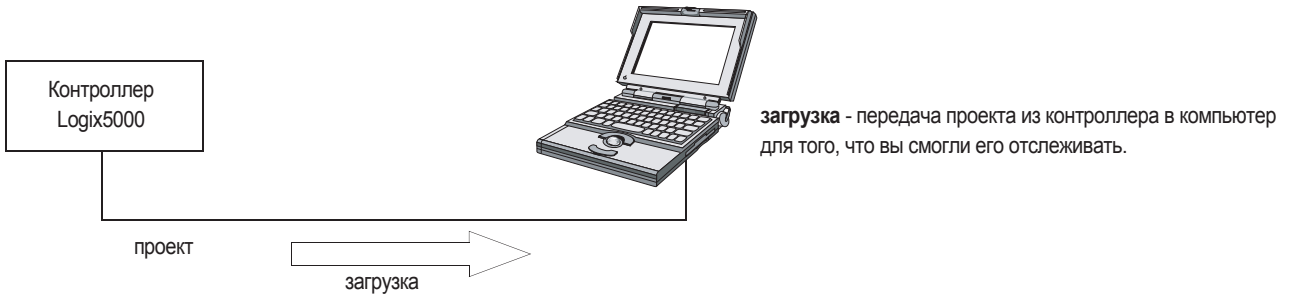


3. Выберите

рабочий режим контроллера



## Если на вашем компьютере нет проекта, выполняемого на контроллере...



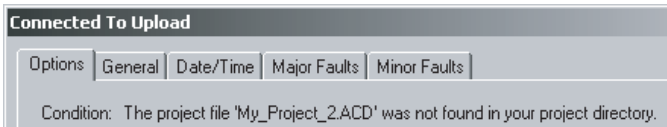
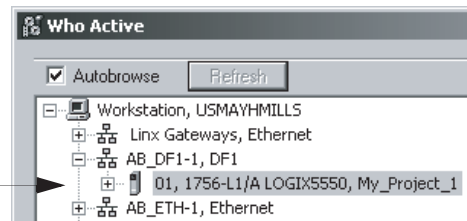
1. Определите путь к контроллеру:

а. Щелкните на

б. Выберите контроллер.

- Чтобы открыть уровень, щелкните на значке +.
- Если контроллер уже выбран, убедитесь что это требуемый контроллер.

2. Выберите



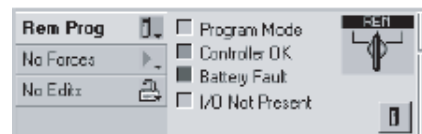
3. Создайте файл проекта на вашем компьютере:

а. Выберите

б. Выберите , а затем



рабочий режим контроллера



## **Для заметок**

## Программирование проекта в режиме онлайн

### Использование этой главы

Данная глава используется при редактировании логики, когда контроллер продолжает управлять вашим механизмом или процессом.

При:	См. стр.
Редактировании логики в режиме онлайн	6-1
Окончательном принятии всех правок в программе	6-5

### Редактирование логики в режиме онлайн

Редактирование в режиме онлайн позволяет вам изменять логику во время работы механизма или процесса.

#### ВНИМАНИЕ



Будьте особенно внимательны при редактировании релейной логики в режиме онлайн. Ошибки могут привести к травмированию персонала и повреждению оборудования. Перед редактированием в режиме онлайн:

- Оцените, как система отреагирует на изменения.
- Доведите изменения до сведения персонала.

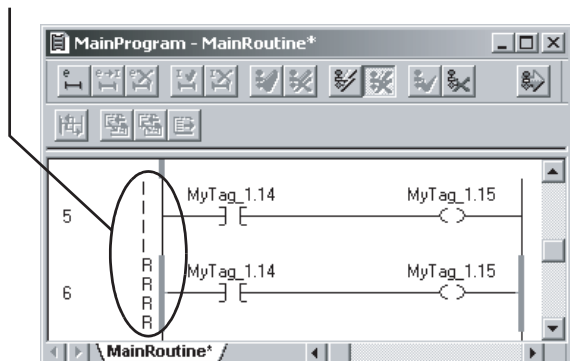
#### ВАЖНО

Когда вы редактируете ПФС в режиме онлайн:

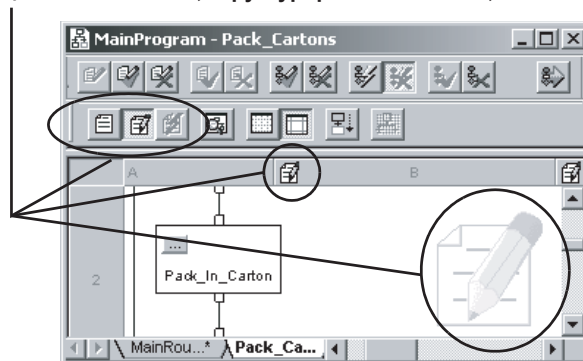
- ПФС перегружается на начальный шаг.
- Все сохраняемые действия отключаются.





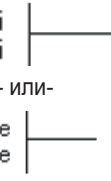





Когда вы осуществляете редактирование в режиме онлайн, RSLogix 5000 использует специальные маркеры для демонстрации состояния процесса редактирования:

Релейная логика



Функциональный блок, структурированный текст, ПФС

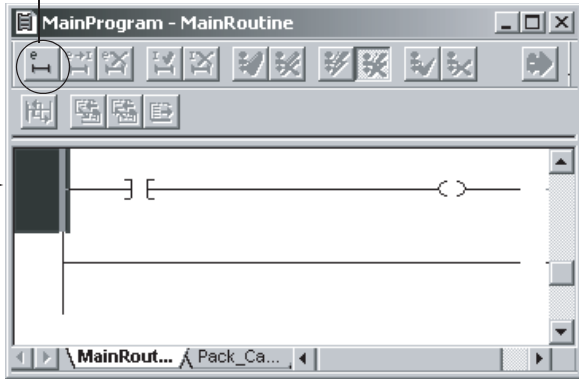


Этот маркер	Означает	Описание								
Релейная логика Функциональный блок Структурированный текст ПФС	 - или - 	<p>Оригинальная логика</p> <p>В режиме онлайн, когда RSLogix 5000 продолжает показывать оригинальную логику, вы редактируете копию алгоритма (отложенное редактирование). Зеленая граница или боковая полоса показывают, какая логика выполняется контроллером в данный момент.</p> <p>В языке функциональных схем, структурированного текста или ПФС, для переключения между различными окнами используются кнопки, расположенные над процедурой.</p>								
										
Релейная логика Функциональный блок Структурированный текст ПФС	 - или - 	<p>Отложенное редактирование</p> <p>Это копия оригинального алгоритма, предназначенная для редактирования. Любые изменения останутся в компьютере, пока вы не согласитесь с результатами редактирования.</p> <ul style="list-style-type: none"> <li>• На языке релейной логики вы редактируете отдельные цепочки в процедуре.</li> <li>• На языке функциональных схем, структурированного текста или ПФС вы редактируете процедуру полностью.</li> </ul>								
										
Релейная логика Функциональный блок Структурированный текст ПФС	 - или - 	<p>Контроль редактирования</p> <p>Когда вы соглашаетесь с результатами отложенного редактирования, программное обеспечение загружает их в контроллер и помечает их как контроль редактирования, а контроллер продолжает выполнять исходный алгоритм. Затем вы вручную задаете выполнение отредактированного алгоритма или возвращаетесь к исходному алгоритму (тестировать или не тестировать правки).</p>								
		<table border="1"> <thead> <tr> <th>Если вы:</th> <th>То:</th> </tr> </thead> <tbody> <tr> <td>Тестируете правки</td> <td> <ul style="list-style-type: none"> <li>• Выполнение переключается на тестирование правок. (выполняются все правки).</li> <li>• Выводы в исходном алгоритме сохраняют свои последние состояния, если только не затрагиваются сделанными правками (или другим алгоритмом).</li> <li>• В ПФС схемы перегружаются на начальные шаги и сохраняемые действия выключаются.</li> </ul> </td> </tr> <tr> <td>Не тестируете правки</td> <td> <ul style="list-style-type: none"> <li>• Выполнение переключается назад на исходный алгоритм.</li> <li>• Выводы в исходном алгоритме сохраняют свои последние состояния, если только не затрагиваются сделанными правками (или другим алгоритмом).</li> <li>• В ПФС схемы перегружаются на начальные шаги и сохраняемые действия выключаются.</li> </ul> </td> </tr> <tr> <td>Транслируете правки</td> <td>Сделанные правки непрерывно заменяют исходный алгоритм.</td> </tr> </tbody> </table> <p>На языке релейной логики, если вы уничтожаете цепочку, программное обеспечение немедленно помечает ее как тестируемую правку (прописное <input type="checkbox"/>)</p>	Если вы:	То:	Тестируете правки	<ul style="list-style-type: none"> <li>• Выполнение переключается на тестирование правок. (выполняются все правки).</li> <li>• Выводы в исходном алгоритме сохраняют свои последние состояния, если только не затрагиваются сделанными правками (или другим алгоритмом).</li> <li>• В ПФС схемы перегружаются на начальные шаги и сохраняемые действия выключаются.</li> </ul>	Не тестируете правки	<ul style="list-style-type: none"> <li>• Выполнение переключается назад на исходный алгоритм.</li> <li>• Выводы в исходном алгоритме сохраняют свои последние состояния, если только не затрагиваются сделанными правками (или другим алгоритмом).</li> <li>• В ПФС схемы перегружаются на начальные шаги и сохраняемые действия выключаются.</li> </ul>	Транслируете правки	Сделанные правки непрерывно заменяют исходный алгоритм.
Если вы:	То:									
Тестируете правки	<ul style="list-style-type: none"> <li>• Выполнение переключается на тестирование правок. (выполняются все правки).</li> <li>• Выводы в исходном алгоритме сохраняют свои последние состояния, если только не затрагиваются сделанными правками (или другим алгоритмом).</li> <li>• В ПФС схемы перегружаются на начальные шаги и сохраняемые действия выключаются.</li> </ul>									
Не тестируете правки	<ul style="list-style-type: none"> <li>• Выполнение переключается назад на исходный алгоритм.</li> <li>• Выводы в исходном алгоритме сохраняют свои последние состояния, если только не затрагиваются сделанными правками (или другим алгоритмом).</li> <li>• В ПФС схемы перегружаются на начальные шаги и сохраняемые действия выключаются.</li> </ul>									
Транслируете правки	Сделанные правки непрерывно заменяют исходный алгоритм.									

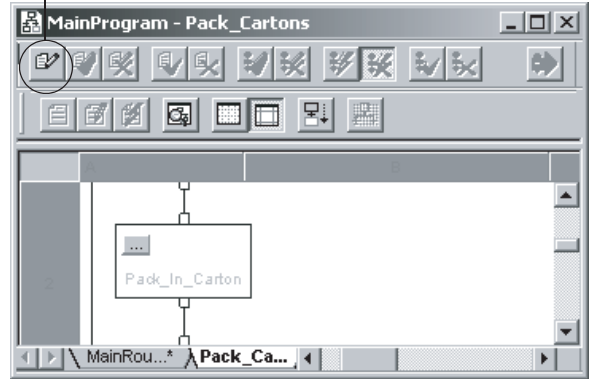
## Запуск отложенного редактирования

1. Для релейной логики щелкните (выберите) цепочку, которую вы хотите редактировать.

2. Запустите отложенное редактирование.



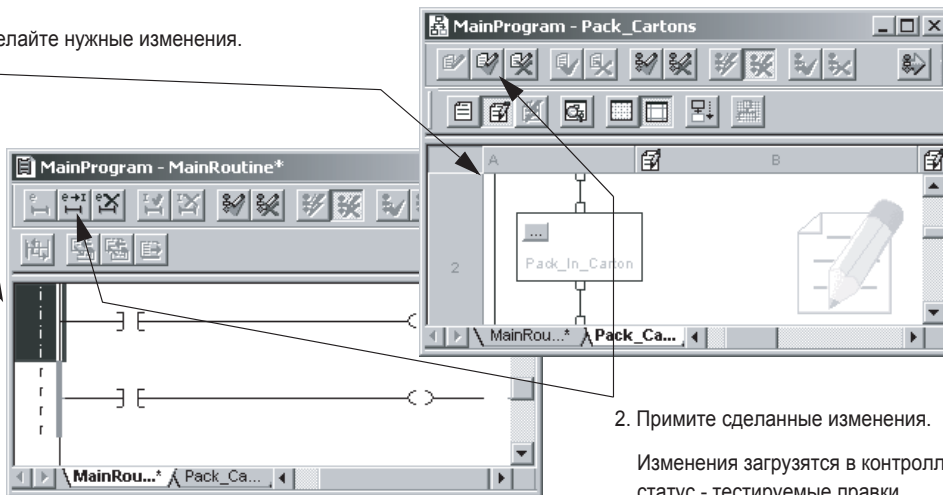
Релейная логика



Функциональная схема, структурированный текст, ПФС

## Редактирование и принятие правок

1. Сделайте нужные изменения.



Функциональная схема, структурированный текст, ПФС

2. Примите сделанные изменения.

Изменения загрузятся в контроллер и получат статус - тестируемые правки.

Релейная логика

### Тестирование правок

1. Проверьте правки на предмет их независимого выполнения.

2. Yes - тестирование правок.

3. Если тестирование не прошло NOT OK, переключите выполнение назад на исходный алгоритм (не тестировать правки). Чтобы сделать изменения, вновь запустите отложенное редактирование.

### Трансляция и сохранение правок


1. Трансляция правок.

Правки заменяют исходный алгоритм.

2. Сохраните проект.

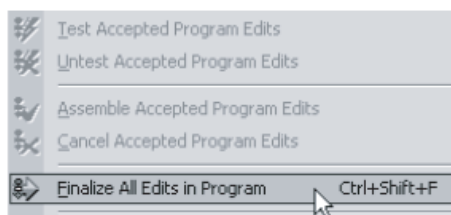


## Окончательное принятие всех правок в программе

 RSLogix 5000  
версии 13.0 или выше

Опция *Finalize All Edits in Program* (Окончательное принятие поправок к программе) позволяет проводить правки в режиме онлайн без тестирования этих правок.

Окончательное принятие поправок к программе



### ВНИМАНИЕ



Будьте очень внимательны используя эту опцию при редактировании в режиме онлайн. Ошибки могут привести к травмированию персонала и повреждению оборудования. Перед редактированием в режиме онлайн:

- Оцените как система будет реагировать на изменения.
- Предупредите персонал об изменениях.

Когда вы выбираете *Finalize All Edits in Program*:

- Все правки в программе (отложенное редактирование и тестирование) немедленно загружаются в контроллер и начинают выполняться.
- Исходный алгоритм удаляется из контроллера.
- Выходные данные, которые были в исходном алгоритме сохраняют свои последние значения, если не используются новым алгоритмом (или другой логикой).
- Если ваши правки включают в себя ПФС:
  - ПФС перегружается на начальный шаг.
  - Сохраненные действия отключаются.

Для использования опции *Finalize All Edits in Program*:

1. Включите отложенное редактирование.
2. Сделайте изменения.
3. Выберите *Finalize All Edits in Program*.

**Для заметок:**

## Устранение неполадок в контроллере

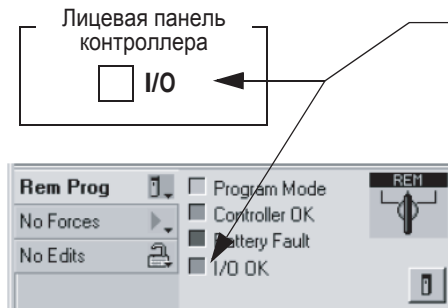
### Использование этой главы

Используйте эту главу, если вы хотите получить основную диагностическую информацию о вашей системе и провести диагностику задач.

Если:	То:	См. стр.
Существует проблема с устройствами в вашей системе, возможно, нарушена связь с модулем ввода/вывода.	Диагностируйте связь с устройствами ввода/вывода	7-2
Ваш процесс неожиданно остановился из-за основной ошибки контроллера.	Сбросьте основную ошибку	7-4
Вы хотите найти определенный элемент (тег, инструкцию и т.д.) внутри проекта.	Найдите проект	7-5
Вы хотите просмотреть проект на предмет поиска определенного элемента (тег, инструкцию и т.д.).	Просмотрите логику	7-7
Вы хотите: <ul style="list-style-type: none"> <li>• Заменить входные данные</li> <li>• Заменить логику</li> <li>• Проверить соединение с выходным устройством.</li> </ul>	Форсируйте значение ввода/вывода	7-8
Вы хотите отобразить данные для одного или более тегов за заданный период.	Создайте и запустите построение тренда (гистограммы)	7-11
Вы хотите посмотреть время сканирования задачи или программы.	Просмотрите время сканирования	7-13

## Диагностирование связи с устройствами ввода/вывода

Если существует проблема с устройствами в вашей системе, возможно, это потеря связи с модулем ввода/вывода.



Состояние связи ввода/вывода

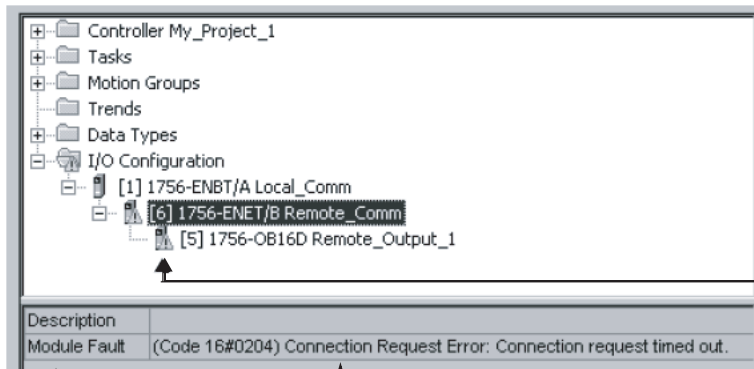
**Если:** То:

Отключено Возможные варианты:

- Отсутствуют модули в конфигурации ввода/вывода контроллера
- Контроллер не содержит проект (память контроллера пуста).

Не мигающий зеленый Контроллер имеет связь со всеми модулями из конфигурации ввода/вывода.

Мигающий зеленый Один или более из модулей конфигурации ввода/вывода контроллера не отвечают.



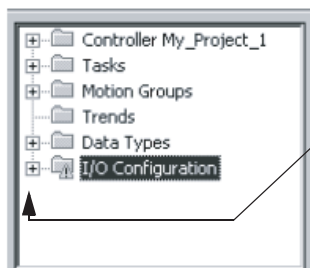
Значок на модуле означает, что у контроллера нет связи с этим модулем.

**соединение** - связь между двумя устройствами, как, например, связь контроллера с модулем ввода/вывода, терминалом PanelView или другим контроллером. Контроллеры Logix5000 используют соединения для связи с модулями ввода/вывода из своей конфигурации.

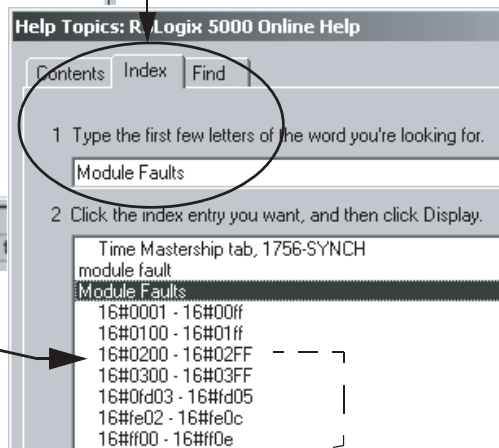
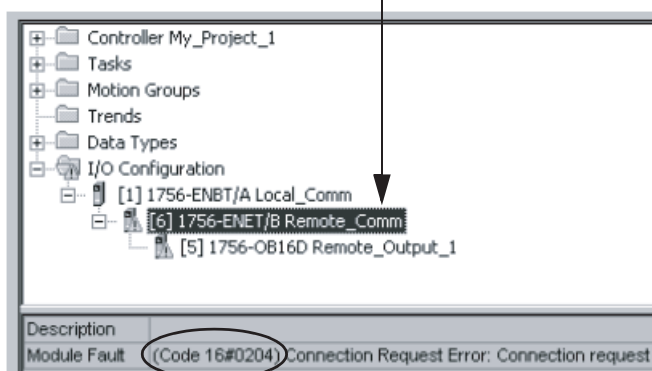
**Ошибка модуля** - нарушена связь с модулем.

**СОВЕТ**

Проверьте в первую очередь модуль связи.  
Сбой коммуникационного модуля влияет на все модули под ним.



1. Войдите в режим контроллера "онлайн".
2. Если необходимо, щелкните на значке + дерева конфигурации ввода/вывода, чтобы увидеть поврежденные модули.
3. Выберите поврежденный модуль.
4. Выберите *Help => Contents*.
5. Щелкните на закладке *Index* и введите с клавиатуры "module faults".

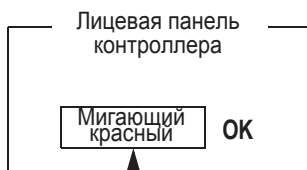


6. Выберите информацию, соответствующую вашей ошибке, и выберите

Module Faults: 16#0200 - 16#02ff		
Code:	String:	Explanation and Possible Causes/Solutions:
16#0203	Connection timed out.	The connection to this module has been interrupted causing a loss of communication. <ul style="list-style-type: none"> <li>Ensure that the module has not been removed and is still functioning and is receiving power. For FLEX I/O modules, ensure that the correct terminal block is in use.</li> <li>Ensure that the network connection to this module has not been interrupted.</li> <li>Call <a href="#">Technical Support</a></li> </ul> <p><b>Note:</b> If a connection to an output module times out and the output module supports Fault Mode and the output module is still functioning, its outputs will transition to the configured Fault Mode.</p>
16#0204	Connection Request Error: Connection request timed out.	The controller is attempting to make a connection to the module and the module is not responding. The controller is not able to communicate with the module. <ul style="list-style-type: none"> <li>Ensure that the module has not been removed and is still functioning and is receiving power. For FLEX I/O modules, ensure that the correct terminal block is in use. Ensure you have entered the correct slot number.</li> </ul>

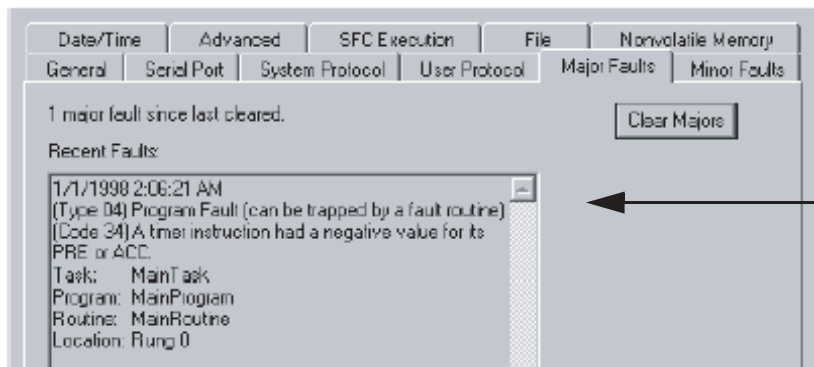
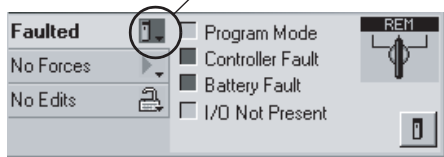
## Сброс основной ошибки

Если ваш процесс неожиданно остановился, возможно, имеет место основная ошибка контроллера.



**Основная ошибка** - контроллер обнаружил условие ошибки, требующее остановку работы.

1. Перейдите в режим онлайн.
2. Выберите *Go To Faults* (*Переход к ошибкам*).



3. Используйте эту информацию для исправления ошибки.

Более подробная информация о кодах ошибок дана в публикации *"Logix5000 Controllers System Reference"*, 1756-QR107.

4. После устранения причины ошибки выберите **Clear Majors**

## Поиск в проекте

Вы можете найти элемент логики (тег, инструкцию, комментарий и т.д.) основываясь на следующих параметрах поиска:

Чтобы найти:	Задайте:	Пример:
тег	Полное или частичное имя	MyTag_1
комментарий/описание	Текст используемый в комментарии/описании	fan
инструкцию	Мнемонику инструкции	OPE
Инструкцию и тег	Мнемонику и тег	OPE MyTag_1

### Поиск всех экземпляров тега, инструкции и т.д.

1. Откройте проект RSLogix 5000 в котором вы хотите искать.
2. Выберите *Search => Find*.
3. Задайте критерий поиска:

а. Введите с клавиатуры символы для поиска.  
Для просмотра тега щелкните на **...**  
выберите тег и выберите **OK**  
Чтобы выбрать разряд щелкните здесь.

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

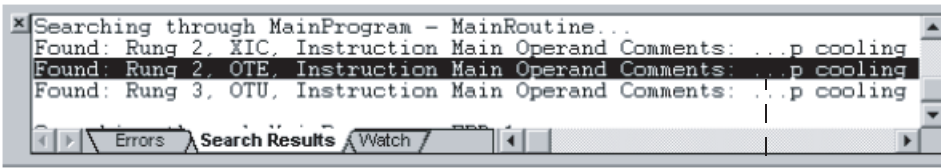
б. Выберите *Text Only* (Только текст).  
в. Выберите *All Routines* (Все процедуры).  
г. Выберите язык и опцию для поиска.  
Для вывода на экран этого раздела диалогового окна выберите **Find Within >>**

4. Выберите **Find All**

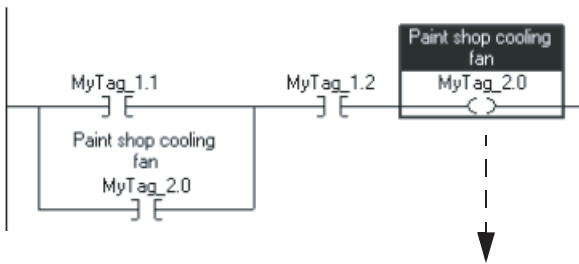
```

x Searching through MainProgram - MainRoutine...
Found: Rung 2, XIC, Instruction Main Operand Comments: ...p cooling
Found: Rung 2, OTE, Instruction Main Operand Comments: ...p cooling
Found: Rung 3, OTU, Instruction Main Operand Comments: ...p cooling
  
```

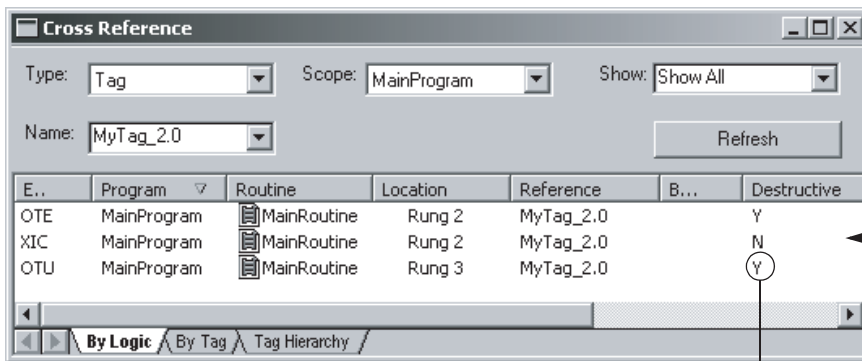
Переход к инструкции



1. Для перехода к инструкции дважды щелкните на ней.



2. Для просмотра список перекрестных ссылок на тег, щелкните правой клавишей мыши и выберите Go To Cross Reference... (Переход к перекрестной ссылке).



3. Для перехода к инструкции дважды щелкните на ней.

"Y" означает, что данная инструкция изменяет значение данного тега.



## Просмотр релейной логики

Для просмотра алгоритма процедуры для поиска заданного элемента (инструкции, элемента, тега, комментария и т.д.) используйте окно Browse Logic (Просмотр логики).



RSLogix 5000  
версии 13.0 или выше

1. В RSLogix 5000 выберите Search (Поиск) => Browse Logic (Просмотр логики).

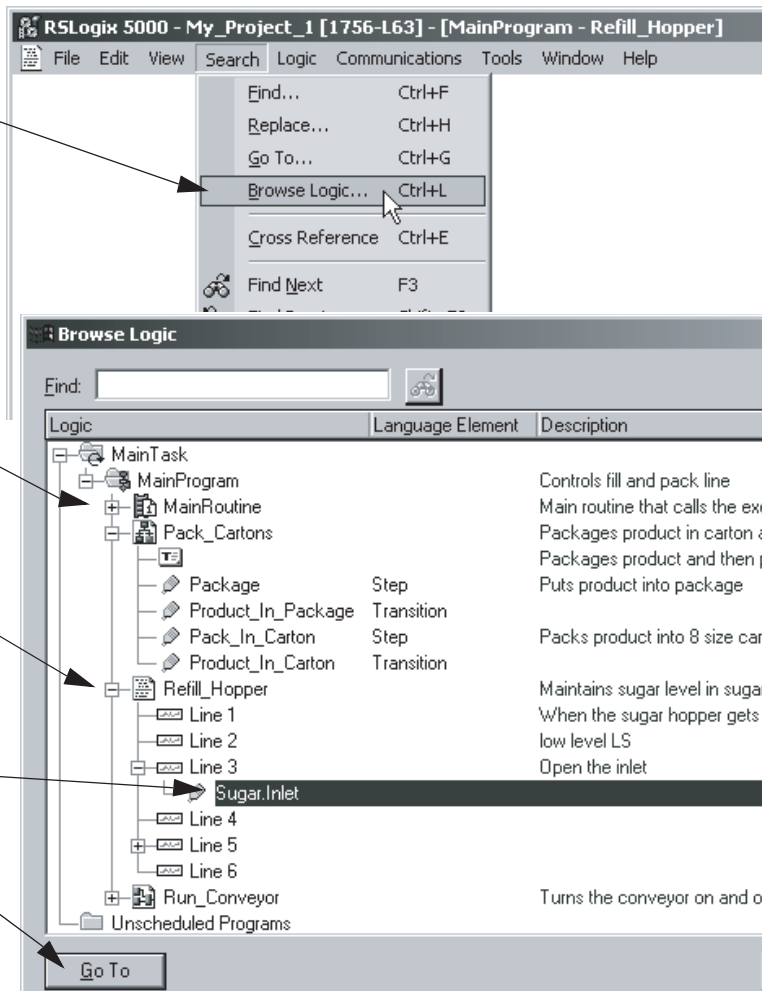
2. Для раскрытия окна и просмотра компонентов, либо:

- Дважды щелкните на входе.
- Щелкните на значке +.
- Щелкните правой клавишей мыши на входе и выберите Expand All (Раскрыть все).

3. Чтобы свернуть окно входа и спрятать содержание, либо:

- Дважды щелкните на входе.
- Щелкните на значке "-".

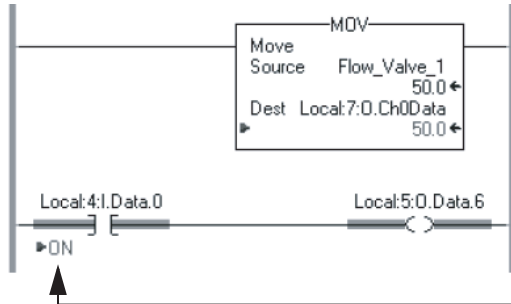
4. Для перехода к положению элемента в логике, выберите этот элемент и выберите Go To (Перейти к).



## Форсировка значения ввода/вывода

Используйте форсировку для замены входных данных или логики, если вам необходимо:

- протестировать и отладить логику
- проверить связь с устройством вывода
- временно продолжить процесс при выходе из строя устройства ввода

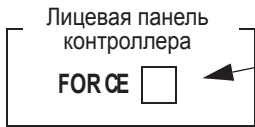


**форсировка** - замена значения, получаемого от устройства ввода или логики.

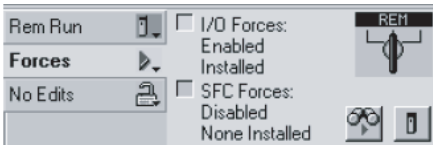
- Форсировка входного тега подменяет значение от входного устройства.
- Форсировка выходного тега меняет логику и посылает это форсированное значение в выходное устройство.



Когда работает форсировка, (включено), рядом с форсируемым элементом появляется значок ►



Состояние форсировок вводов/вывода.



Если:	То:
Off (выключено)	<ul style="list-style-type: none"> <li>• Нет тегов, содержащих форсированные значения ввода/вывода.</li> <li>• Форсировка ввода/вывода не работает (отключена).</li> </ul>
Мигающий янтарный	<ul style="list-style-type: none"> <li>• Один или более тегов содержат форсированное значение.</li> <li>• Форсировка ввода/вывода работает (включена).</li> <li>• Когда вы включили форсировку ввода/вывода, все существующие форсировки ввода/вывода имеют место.</li> </ul>
Постоянный янтарный	<ul style="list-style-type: none"> <li>• Форсировка ввода/вывода работает (включена).</li> <li>• Форсированные значения могут иметь место, а могут и нет.</li> <li>• Когда вы устанавливаете (добавляете) форсировку, она начинает действовать немедленно.</li> </ul>

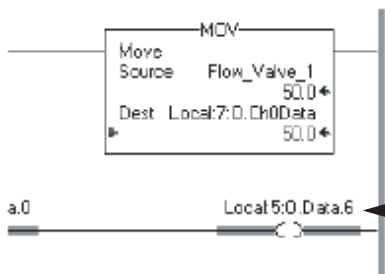
Если вы хотите:	То:
Заменить значение	Установите форсировку ввода/вывода (форсируйте значение ввода/вывода).
Отменить отдельную форсировку, но оставить остальные форсировки включенными и действующими.	Удалите отдельную форсировку
Отменить все форсировки ввода/вывода, но оставить форсировку ввода/вывода в проекте.	Отключите все форсировки ввода/вывода

**ВНИМАНИЕ**

Форсировка может привести к неожиданному движению механизмов, что может повлечь за собой травмы персонала. Перед использованием форсировки определите, как она повлияет на ваш механизм или процесс, и удалите людей из зоны действия механизмов.

- Разрешение форсировки ввода/вывода вызывает изменение входных, выходных, произведенных или потребленных значений.
- Удаление отдельной форсировки может, тем не менее, сохранить состояние разрешения форсировок.
- Если форсировки разрешены, и вы устанавливаете форсировку, она действует немедленно.

### Включение форсировки ввода/вывода



1. Войдите в режим онлайн работы с контроллером и откройте процедуру, содержащую тег, который вы хотите форсировать.
2. Щелкните правой клавишей мыши на этом теге и выберите *Monitor...*
3. Если необходимо, щелкните на значке + данного тега для просмотра значения, которое вы хотите форсировать (например, значение BOOL тега DINT).

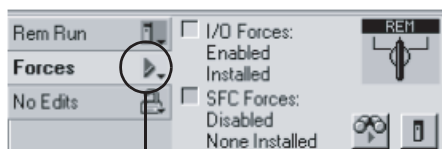
Tag Name	Value	Force Mask
+ Local:4:C	{...}	{...}
- Local:4:I	{...}	Forced
+ Local:4:I.Fault	2#00...	
- Local:4:I.Data	2#0...	2#...._...
- Local:4:I.Data.0	1	1
- Local:4:I.Data.1	0	

4. Установите значение форсировки:

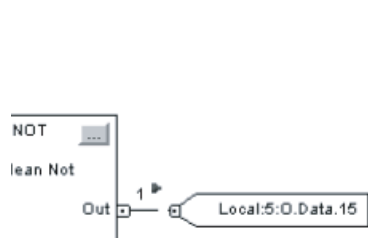
#### Для форсировки: Сделайте следующее:

Значения типа BOOL	Щелкните правой клавишей мыши на теге и выберите <i>Force ON</i> или <i>Force OFF</i>
Значение типа integer или REAL	В колонке <i>Force Mask</i> для данного тега введите с клавиатуры значение, на которое вы хотите форсировать и нажмите клавишу Enter.

5. Выберите *I/O Forcing => Enable All I/O Forces*. И выберите  (yes - включает форсировки ввода/вывода).



## Удаление отдельной форсировки

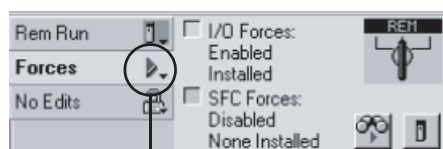


1. Войдите в режим онлайн работы с контроллером и откройте процедуру, содержащую тег, который вы хотите форсировать.
2. Щелкните правой клавишей мыши на этом теге и выберите *Monitor...*
3. Если необходимо, щелкните на значке + данного тега для просмотра его членов (например, значение BOOL тега DINT).

Tag Name	Value	Force Mask
+ Local:4:C	{...}	{...}
- Local:4:I	{...}	Forced
+ Local:4:I.Fault	2#00...	
- Local:4:I.Data	▶ 2#0...	2#...._...
Local:4:I.Data.0	▶ 1	1
Local:4:I.Data.1	0	

4. Щелкните правой клавишей мыши на данном теге и выберите *Remove Force*.

## Выключение всех форсировок ввода/вывода

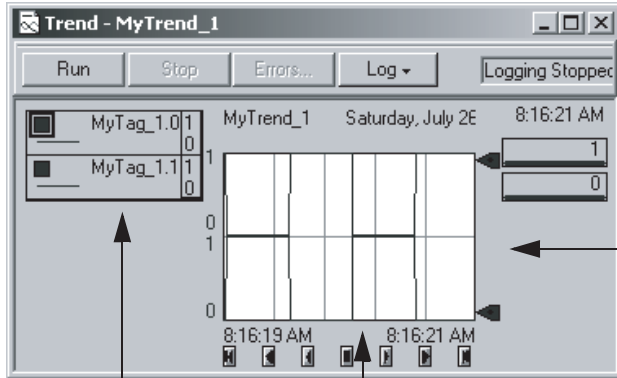


1. Войдите в режим онлайн работы с контроллером .

2. Выберите *I/O Forcing => Disable All I/O Forces*. И выберите  (yes- отключает форсировки ввода/вывода).

## Создание и запуск тренда (гистограммы)

Тренды позволяют вам просматривать данные отобранного тега за определенный период времени в графическом виде. Данные тега отбираются контроллером и затем выводятся на экран в виде точки (точек) на графике тренда.



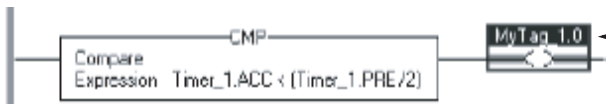
тренд - выборка заданных тегов и демонстрация этих данных на графическом дисплее.

Теги, которые вы хотите просмотреть

Значения тегов

время

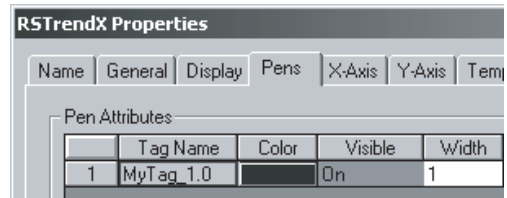
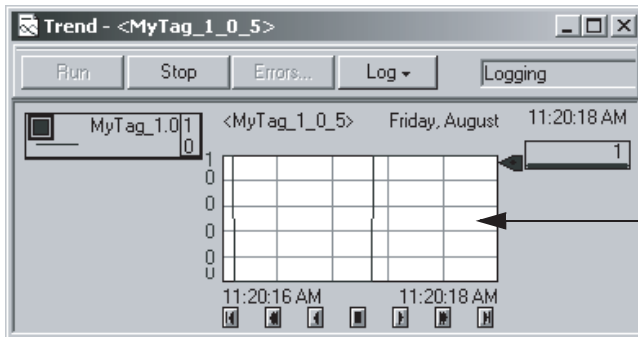
### Запуск тренда для тега



Щелкните правой клавишей мыши на первом теге, для которого вы хотите получить тренд и выберите *Trend...*

### Добавление дополнительных тегов к тренду

1. Щелкните правой клавишей мыши на графике и выберите *Chart Properties (Свойства графика)*.
2. Щелкните на закладке *Pens*.

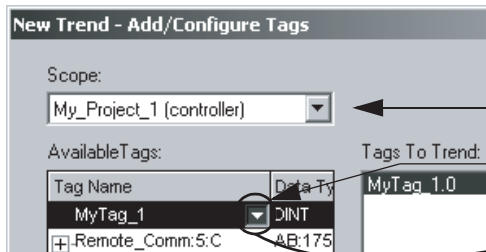


3. Выберите **Add/Configure Tags**

4. Выберите тег, который вы хотите добавить и нажмите **Add -->**

Для изменения масштаба выберите score (Масштаб).

Для выбора разрядности щелкните на ▼

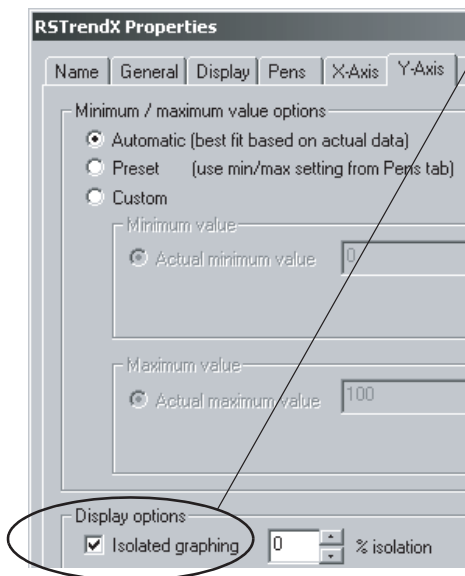


0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

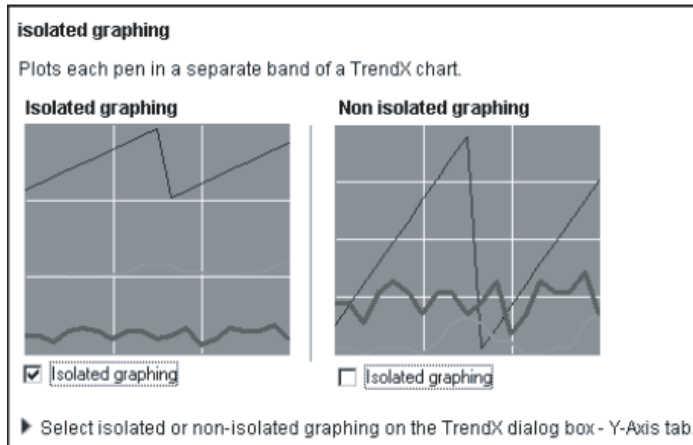
5. Когда вы уже добавили нужный тег, щелкните на **OK**

Добавление дополнительных тегов к тренду (продолжение)

6. Щелкните на закладке Y-Axis.



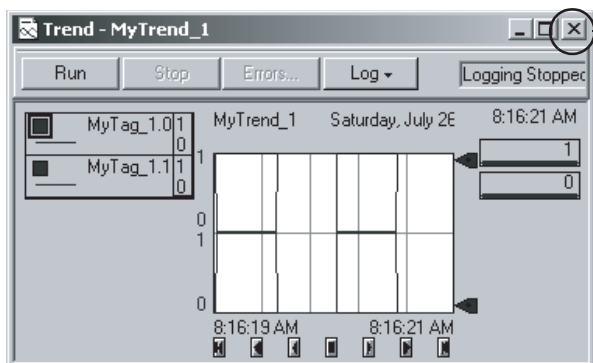
7. Выберите тип графика.



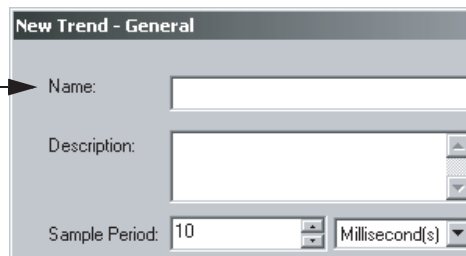
8. выберите

9. Для получения тренда выберите

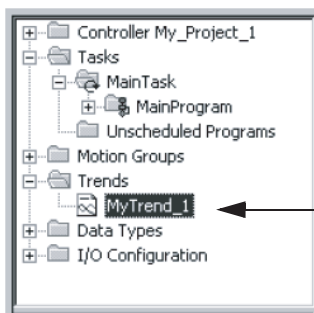
Не обязательная опция - сохранение тренда



1. Когда вы выбрали тренд, вы можете его сохранить для последующего использования.



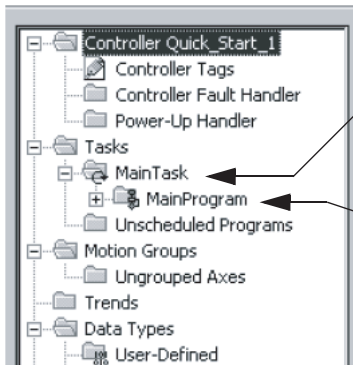
2. Введите с клавиатуры имя для данного тренда и выберите



тренд

## Просмотр времени сканирования

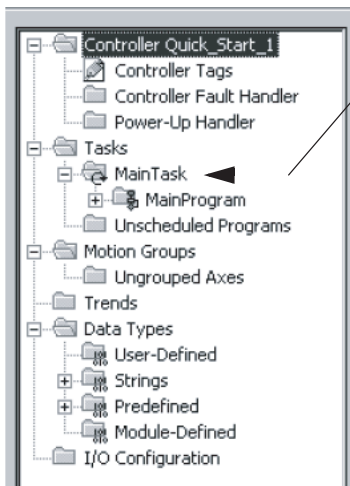
Контроллер Logix5000 использует два типа времени сканирования, каждый из которых служит своим целям.



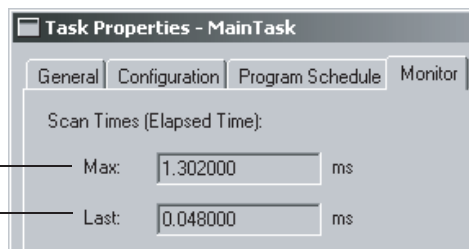
**затраченное время (время сканирования задачи) (elapsed time (task scan time))** - время, которое проходит с момента пуска задачи до ее окончания в миллисекундах. Затраченное время на задачу включает время, которое потратила задача, прерываясь на обслуживание связи или других задач.

**время выполнения (программное время сканирования) (execution time)** - время на выполнение логики из какой либо программы (главной процедуры и всех подпрограмм, вызываемых главной процедурой) в миллисекундах. Время сканирования программы включает в себя только время выполнения данной логики. Прерывания *не включаются*.

### Просмотр времени сканирования задачи



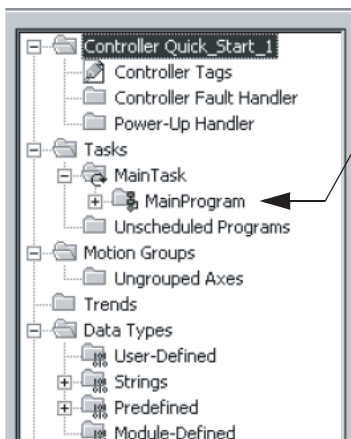
1. Щелкните правой клавишей мыши и выберите *Properties*.
2. Щелкните на закладке *Monitor*.



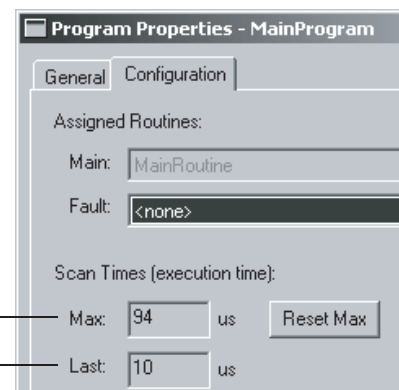
время, затраченное на последнее выполнение этой задачи

максимальное время выполнения этой задачи

### Просмотр времени сканирования программы



1. Щелкните правой клавишей мыши и выберите *Properties*.
2. Щелкните на закладке *Configuration*.



максимальное время выполнения этой программы

время, затраченное на последнее выполнение этой программы

**Для заметок:**



**A****alias tags (теги псевдонимы)**

use of (использование) 1-13

**array (массив)**

create (создание) 2-5

organize (организация) 3-22

use of (использование) 2-5

**ASCII text (текст ASCII)**enter logic using (ввод используемой логики)  
3-2**assume data available indicator (указатель**

«предположение о наличии данных»)

use of (использование) 3-9

**B****BOOTP**

use of (использование) 5-2

**browse (просмотр)**

logic (логики) 7-7

**C****clear (сброс)**

major fault (основной ошибки) 7-4

**comment (комментарий)**add to function block diagram (добавить к  
функциональной блок-схеме) 4-7

add to rung (добавить к цепочке) 4-4, 4-5

add to SFC (добавить к ПФС) 4-7

add to structured text (добавить к  
структурированному тексту) 4-9

search for (поиск) 7-5

**communicate (связываться)**with controller via EtherNet/IP network (с  
контроллером через сеть EtherNet/IP) 5-2with controller via serial cable (с контроллером  
через кабель последовательного порта) 1-  
15**communication (обмен данными)**

fault (ошибка) 7-2

**configure (конфигурирование)**

controller (контроллера) 1-2, 2-11

driver for EtherNet/IP communication (драйвера  
для связи через EtherNet/IP) 5-2driver for serial communication (драйвера для  
связи через последовательный порт) 1-15

I/O module (модуля ввода/вывода) 1-4, 2-12

task (задачи) 2-2

trend (тренда) 7-11

**continuous task (непрерывная задача)**

execution (выполнение) 2-2

**controller (контроллер)**communicate via EtherNet/IP network (связь  
через сеть EtherNet/IP) 5-2communicate via serial cable (связь через  
кабель последовательного порта) 1-15

configure (конфигурирование) 1-2, 2-11

download project (загрузка проекта) 1-17

faulted (сброшенный) 7-4

go online with (войти в режим онлайн) 5-6

mode (режим) 1-19

monitor (контроль) 5-8

monitor execution (выполнение контроля) 5-6

revision (версия) 1-17

**controller organizer (организатор контроллера)**add I/O module (добавить модуль ввода/  
вывода) 1-4

navigate (навигация) 1-2

open routine (открытие процедуры) 1-7

**controller-score tags (теги в видимости  
контроллера)**

when to use (когда использовать) 2-3

**create (создание)**

program (программы) 2-3

project (проекта) 1-2

routine (процедуры) 2-7

sheet (листа) 3-9

text box (текстового окна) 4-7

trend (тренда) 7-11

**D****data (данные)**

I/O module (модуля ввода/вывода) 1-5

trend (тренда) 7-11

**description (описание)**

rung (цепочки) 4-4, 4-5

search for (поиска) 7-5

tag (тега) 4-2

user-defined data type (пользовательского типа  
данных) 4-2**document (документирование)**function block diagram (функциональной блок-  
схемы) 4-7

rung (цепочки) 4-4, 4-5

SFC (ПФС) 4-7

structured text (структурированного текста) 4-9

tag (тега) 4-2

user-defined data type (пользовательского типа  
данных) 4-2**download (загрузка)**

project (проекта) 1-17

**driver (драйвер)**configure for EtherNet/IP communication  
(конфигурирование для связи через

EtherNet/IP) 5-2

## E

### elapsed time (время выполнения)

task (задачи) 7-13

### enter

function block diagram (функциональной блок-схемы) 3-9

ladder logic (релейной логики) 3-2

logic while online (логики в режиме онлайн) 6-1, 6-5

rung comment (комментария к цепочке) 4-4

SFC (ПФС) 3-16

structured text (структурированного текста) 3-14

comments (комментариев) 4-9

### errors (ошибки)

check routine for (проверка процедуры) 3-20

### EtherNet/IP network (сеть EtherNet/IP)

assign IP address (присваивание адреса IP) 5-2

communicate with controller (связь с контроллером) 5-2

### execution (выполнение)

choose controller mode (выбор режима контроллера) 1-19

task (задачи) 2-2

time (время) 7-13

### export (экспорт)

ladder logic (релейной логики) 3-6

rung comment (комментария к цепочке) 4-5

## F

### Faceplate (разметочная панель)

add (добавление) 3-12

### fault (ошибка)

controller (контроллера) 7-4

I/O module (модуля ввода/вывода) 7-2

### File (файл)

See array (См. массив)

### finalize all edits in program (окончательное принятие поправок к программе) 6-5

### find (поиск)

See search (См. Search)

### firmware (микропрограммное обеспечение)

update during load (обновление при загрузке) 1-17

### force (форсировка)

I/O value (значения ввода/вывода) 7-8

### function block diagram (функциональная блок-схема)

create sheet (создание листа) 3-9

document (документирование) 4-7

edit online (редактирование в режиме онлайн) 6-1, 6-5

enter (ввод) 3-9

resolve loop (разрешение цикла) 3-9

use for (использование) 2-7

### function block instruction (инструкция функциональной блок-схемы)

use of faceplate (использование разметочной панели) 3-12

## H

### Histogram (гистограмма)

See trend (См. тренд)

## I

### I/O device (устройство ввода/вывода)

access data (доступ к данным) 1-5

### I/O module (модуль ввода/вывода)

add to project (добавление к проекту) 1-4

address format (формат адреса) 1-5

communication failure (сбой связи) 7-2

configure (конфигурирование) 1-4, 2-12

faulted (повреждение) 7-2

force value (форсировка значения) 7-8

### import (импорт)

ladder logic (релейной логики) 3-6

rung comment (комментария к цепочке) 4-5

### instruction (инструкция)

search for (поиск) 7-5

### IP address (адрес IP)

assign to module (присваивание модулю) 5-2

## L

### ladder logic (релейная логика)

add rung comment (добавление комментария к цепочке) 4-4, 4-5

edit online (редактирование в режиме онлайн) 6-1, 6-5

enter (ввод) 8-10

export (экспорт) 8-14

import (импорт) 8-14

use for (использование) 2-7

use of quick keys (использование быстрого доступа) 3-2

## M

### main routine (главная процедура)

assign (присваивание) 2-10

use of (использование) 2-7

**major fault (основная ошибка)**

clear (сброс) 7-4

**mode (режим)**

controller (контроллера) 1-19

**monitor (контроль)**

controller (контроллера) 5-6

project in controller (проекта в контроллере) 5-8

## N

**name (имя)**

guidelines for tag (указания для тега) 3-22

limitations (ограничения) 1-2

## O

**Online (режим онлайн)**

edit logic (редактирование логики) 6-1, 6-5

finalize all edits (окончательное принятие всех правок) 6-5

with controller (контроллера) 5-6

**open (открытие)**

routine (процедуры) 1-7

**operand**

assign (присваивание) 3-18

## P

**pass-through description (сквозное описание) 4-2**

**period (период)**

define for a task (задание для задачи) 2-2

**periodic task (периодическая задача)**

execution (выполнение) 2-2

**program (программа)**

assign main routine (присвоение главной процедуры) 2-10

create (создание) 2-3

finalize all edits (окончательное принятие всех правок) 6-5

scan time (время сканирования) 7-13

**program mode (программный режим) 1-19**

**programming language (язык программирования)**

choose (выбор) 2-7

RSLogix 5000 software (программное обеспечение RSLogix 5000) 2-8

**program-scope tags (теги в видимости программы)**

when to use (когда использовать) 2-3

**project (проект)**

create (создание) 1-2

download (загрузка) 1-17

monitor in controller (проверка в контроллере) 5-6

organize routines (организация процедур) 2-7

upload (выгрузка) 5-8

verify (проверка) 3-20

## Q

**quick keys (быстрый доступ)**

enter ladder logic (ввод релейной логики) 3-2

## R

**Revision (версия)**

controller firmware (микропрограммное обеспечение контроллера) 1-17

**routine (процедура)**

check for errors (проверка на ошибки) 3-20

create (создание) 2-7

create tag (создание тега) 3-18

edit logic online (редактирование логики в режиме онлайн) 6-1, 6-5

import ladder logic (импорт релейной логики) 3-6

open (открытие) 1-7

organize (организация) 2-7

program ladder logic (программирование релейной логики) 3-2

program using a function block diagram (программирование с использованием функциональной блок-схемы) 3-9

program using an SFC (программирование с использованием ПФС) 3-16

program using structured text (программирование с использованием структурированного текста) 3-14

**run mode (режим работы) 1-19**

**rung comment (комментарий к цепочке)**

add (добавление) 4-4, 4-5

export/import (экспорт/импорт) 4-5

## S

**scan time (время сканирования)**

view (просмотр) 7-13

**scope (область видимости)**

choose for tag (выбор для тега) 2-3

guidelines (указания) 3-22

**search (поиск)**

browse (просмотр) 7-7

comments or descriptions (комментариев

или описаний) 7-5  
 instruction (инструкции) 7-5  
 tag (тега) 7-5  
**sequential function chart (последовательная функциональная схема)**  
 See SFC (См. ПФС)  
**serial communication (связь через последовательный порт)**  
 with controller (с контроллером) 1-15  
**SFC (ПФС)**  
 document (документирование) 4-7  
 edit online (редактирование в режиме онлайн) 6-1, 6-5  
 enter (ввод) 3-16  
 use for (использование) 2-7  
**sheet лист**  
 use of (использование) 3-9  
**source protection (защита источника)**  
 use of (использование) 1-7  
**structure (структура)**  
 create (создание) 2-5  
 organize (организация) 3-22  
**structured text (структурированный текст)**  
 document (документирование) 4-9  
 edit online (редактирование в режиме онлайн) 6-1, 6-5  
 enter (ввод) 3-14  
 use for (использование) 2-7  
**subroutine (подпрограмма)**  
 See routine (См. Процедура)

**T**

**tag (тег)**  
 create (создание) 3-18  
 description (описание) 4-2  
 force value (значение форсировки) 7-8  
 format (формат) 3-18  
 guidelines (указания) 3-22  
 I/O module (модуль ввода/вывода) 1-5  
 organize (организация) 2-5, 3-22  
 reuse of names (повторное использование имен) 2-3  
 scope (масштаб) 2-3  
 search for (поиск) 7-5  
 trend value (значение тренда) 7-11  
**task (задача)**  
 configure (конфигурирование) 2-2  
 scan time (время сканирования) 7-13  
**test mode (режим проверки) 1-19**  
**text box (текстовое окно)**  
 add to function block diagram (добавление к

функциональной блок-схеме) 4-7  
 add to SFC ( добавление к ПФС) 4-7  
**trend (тренд)**  
 create and run (создание и запуск) 7-11  
**troubleshoot (диагностика сбоев)**  
 check wiring to output device (проверка соединения с устройством вывода) 7-8  
 communication with I/O module (связь с модулем ввода/вывода) 7-2  
 entire system is shut down (останов всей системы) 7-4  
 override logic (замена логики) 7-8  
 see data history (просмотр предыстории данных) 7-11  
 several devices not responding (не отвечает несколько устройств) 7-2

## U

**Update (обновление)**  
 controller firmware (микропрограммного обеспечения контроллера) 1-17  
**upload (загрузка)**  
 project (проекта) 5-8  
**user-defined data type (пользовательский тип данных)**  
 create (создание) 2-5  
 use of (использование) 2-5

## V

**Verify (проверка)**  
 project (проекта) 3-20







# How Are We Doing?

Your comments on our technical publications will help us serve you better in the future. Thank you for taking the time to provide us feedback.

You can complete this form and mail (or fax) it back to us or email us at [RADocumentComments@ra.rockwell.com](mailto:RADocumentComments@ra.rockwell.com)

Pub. Title/Type Logix5000™ Controllers Quick Start

Cat. No.	1756 ControlLogix®, 1769 CompactLogix™, 1789 SoftLogix™, 1794 FlexLogix™, PowerFlex® 700S with DriveLogix™	Pub. No.	<u>1756-QS001B-EN-P</u>	Pub. Date	<u>March 2004</u>	Part No.	<u>957867-44</u>
----------	--	----------	-------------------------	-----------	-------------------	----------	------------------

Please complete the sections below. Where applicable, rank the feature (1=needs improvement, 2=satisfactory, and 3=outstanding).

<b>Overall Usefulness</b>	1	2	3	How can we make this publication more useful for you?									
<b>Completeness</b> (all necessary information is provided)	1	2	3	Can we add more information to help you?									
				<table border="0" style="width: 100%;"> <tr> <td style="width: 33%;">procedure/step</td> <td style="width: 33%;">illustration</td> <td style="width: 33%;">feature</td> </tr> <tr> <td>example</td> <td>guideline</td> <td>other</td> </tr> <tr> <td>explanation</td> <td>definition</td> <td></td> </tr> </table>	procedure/step	illustration	feature	example	guideline	other	explanation	definition	
procedure/step	illustration	feature											
example	guideline	other											
explanation	definition												
<b>Technical Accuracy</b> (all provided information is correct)	1	2	3	Can we be more accurate?									
				<table border="0" style="width: 100%;"> <tr> <td style="width: 50%;">text</td> <td style="width: 50%;">illustration</td> </tr> </table>	text	illustration							
text	illustration												
<b>Clarity</b> (all provided information is easy to understand)	1	2	3	How can we make things clearer?									
<b>Other Comments</b>				You can add additional comments on the back of this form.									

Your Name \_\_\_\_\_  
 Your Title/Function \_\_\_\_\_  
 Location/Phone \_\_\_\_\_

Would you like us to contact you regarding your comments?  
 No, there is no need to contact me  
 Yes, please call me  
 Yes, please email me at \_\_\_\_\_  
 Yes, please contact me via \_\_\_\_\_

Return this form to: Rockwell Automation Technical Communications, 1 Allen-Bradley Dr., Mayfield Hts., OH 44124-9705  
 Fax: 440-646-3525 Email: [RADocumentComments@ra.rockwell.com](mailto:RADocumentComments@ra.rockwell.com)

PLEASE FASTEN HERE (DO NOT STAPLE)

Other Comments

PLEASE FOLD HERE



NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

PLEASE REMOVE

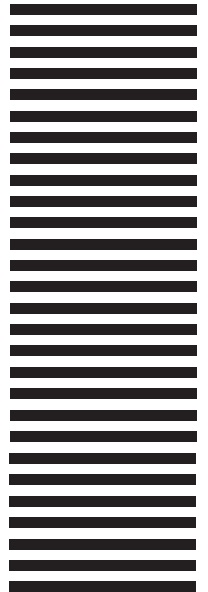
**BUSINESS REPLY MAIL**

FIRST-CLASS MAIL PERMIT NO. 18235 CLEVELAND OH

POSTAGE WILL BE PAID BY THE ADDRESSEE



1 ALLEN-BRADLEY DR  
MAYFIELD HEIGHTS OH 44124-9705





Microsoft, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

## Поддержка Rockwell Automation

Rockwell Automation предоставляет техническую информацию в Интернете, чтобы помочь вам в использовании наших продуктов. По адресу <http://support.rockwellautomation.com> вы найдете технические руководства, базу FAQ, технические заметки и заметки по использованию, образец программного кода и ссылки на пакеты обновления программного обеспечения, а также средство MySupport, которое вы можете настроить под себя для наилучшего использования этих инструментов.

Дополнительно к технической поддержке по телефону при установке, конфигурировании и поиске неисправностей, мы предлагаем программы TechConnect Support. За более подробной информацией вы можете обратиться к местным дистрибьюторам, представителям Rockwell Automation или на сайт <http://support.rockwellautomation.com>.

### Помощь при установке

Если у вас есть проблемы с аппаратным модулем в первые 24 часа после установки, пожалуйста, обратитесь к данному руководству. Вы также можете обратиться в службу поддержки пользователей за первой помощью по наладке и запуску вашего модуля по телефону:

---

США	1.440.646.3223 Понедельник – пятница, 8:00 – 17:00 по восточному времени
Для других стран	Пожалуйста, обращайтесь к местным представителям Rockwell Automation за любой технической поддержкой.

---

### Возврат новых продуктов в связи с неудовлетворительной работой

Компания Rockwell проверяет все свои продукты на предмет полной работоспособности после отгрузки с завода-изготовителя. Однако, если ваш продукт не работает в полном объеме и должен быть возвращен:

---

США	Обратитесь к вашему дистрибьютору. Для организации возврата вы должны сообщить дистрибьютору номер, присвоенный службой поддержки пользователей (Customer Support) (его можно получить по указанному выше номеру телефона).
Для других стран	Обратитесь к местному представителю Rockwell Automation для организации процедуры возврата.

---

ControlNet является торговой маркой ControlNet International.  
DeviceNet является торговой маркой Open DeviceNet Vendor Association.

[www.rockwellautomation.com](http://www.rockwellautomation.com)

#### Corporate Headquarters

Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI, 53202-5302 USA, Tel: (1) 414.212.5200, Fax: (1) 414.212.5201

#### Headquarters for Allen-Bradley Products, Rockwell Software Products and Global Manufacturing Solutions

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444  
Europe: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36-BP 3A/B, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640  
Asia Pacific: Rockwell Automation, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

#### Headquarters for Dodge and Reliance Electric Products

Americas: Rockwell Automation, 6040 Ponders Court, Greenville, SC 29615-4617 USA, Tel: (1) 864.297.4800, Fax: (1) 864.281.2433  
Europe: Rockwell Automation, Brühlstraße 22, D-74834 Elztal-Dallau, Germany, Tel: (49) 6261 9410, Fax: (49) 6261 17741  
Asia Pacific: Rockwell Automation, 55 Newton Road, #11-01/02 Revenue House, Singapore 307987, Tel: (65) 351 6723, Fax: (65) 355 1733



**Allen-Bradley**

**Руководство для ускоренной подготовки к работе  
с семейством программируемых контроллеров Logix5000™**

**Упрощенное руководство**

---

**Связывайтесь с нами теперь по адресу [www.rockwellautomation.com](http://www.rockwellautomation.com)**

Всякий раз, когда вы нуждаетесь в нас, Rockwell Automation осуществляет комплексное использование ведущих марок в промышленной автоматике, включающих элементы управления Allen-Bradley, продукты подачи питания Reliance Electric, механические компоненты подачи питания Dodge и Rockwell Software. Уникальный гибкий подход Rockwell Automation в помощи клиентам достигнуть конкурентного преимущества поддерживается тысячами авторизованными партнерами, дистрибьюторами и системными интеграторами по всему миру.

**Представительство Rockwell Automation в Москве:** 113054, Москва, Большой Строченовский пер., 22/25, Офис 402  
Телефон: (095)956-0464, (095)956-0465; Факс: (095)956-0469; E-mail: [software@rockwell.ru](mailto:software@rockwell.ru), [info@rockwell.ru](mailto:info@rockwell.ru)

**Americas Headquarters,** 1201 South Second Street, Milwaukee, WI 53204, USA, Tel: (1) 414 382-2000, Fax: (1) 414 382-4444

**European Headquarters SA/NV,** avenue Herrmann Debroux, 46, 1160 Brussels, Belgium, Tel: (32) 2 663 06 00, Fax: (32) 2 663 06 40

**Asia Pacific Headquarters,** 27/F Citicorp Centre, 18 Whitfield Road Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

