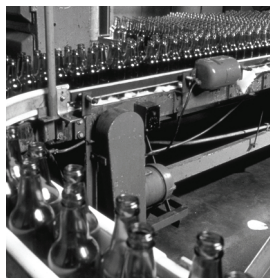


Использование языка функциональных блок-схем в контроллерах Logix5000



Руководство по программированию

**Номера по каталогу 1756 ControlLogix,
1769 CompactLogix, 1789 SoftLogix,
1794 FlexLogix, PowerFlex 700S c
DriveLogix**

Важная информация для пользователей





Рабочие характеристики полупроводникового оборудования отличаются от параметров электромеханического оборудования. Публикация SGI-1.1 Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (Основы безопасности при использовании, установке и обслуживании полупроводниковых устройств), которую можно получить в региональном офисе отдела продаж корпорации Rockwell Automation или в Интернете (<http://www.ab.com/manuals/gi>), описывает некоторые важные различия между полупроводниковым оборудованием и электромеханическими устройствами. Из-за этих различий, а также ввиду широкого разнообразия в применении различных полупроводниковых устройств, персонал, ответственный за работу с указанным оборудованием, должен убедиться, что в каждом конкретном случае такое применение является целесообразным.

Корпорация Rockwell Automation, Inc. не берет на себя ответственность за прямой или косвенный ущерб, возникший при использовании этого оборудования.

Примеры и схемы в данном руководстве приведены исключительно в иллюстративном качестве. Поскольку с любым конкретным устройством связано множество переменных параметров и требований, корпорация Rockwell Automation, Inc. не может принять на себя каких-либо обязательств или ответственности за практическое применение приведенных здесь примеров и схем.

Корпорация Rockwell Automation, Inc. не принимает на себя никаких патентных обязательств в отношении использования информации, схем подключения, оборудования и программного обеспечения, приведенных в данном руководстве.

Полное или частичное воспроизведение содержимого данного документа без письменного разрешения Rockwell Automation, Inc. запрещено. В данном руководстве мы обращаем ваше внимание на вопросы техники безопасности с помощью следующих обозначений.

ПРЕДУПРЕЖДЕНИЕ 	Обозначает информацию о действиях и обстоятельствах, которые могут привести к взрыву в опасных условиях, к травмам или смерти людей, повреждению собственности или экономическому ущербу.
ВАЖНО	Обозначает информацию, наиболее важную для успешной эксплуатации устройства и понимания особенностей его работы.
ВНИМАНИЕ 	Обозначает информацию о действиях и обстоятельствах, которые могут привести к взрыву в опасных условиях, к травмам или смерти людей, повреждению собственности или экономическому ущербу. Данное обозначение помогает определить опасность, избежать опасности и оценить последствия.
ОПАСНОСТЬ ПОРАЖЕНИЯ ЭЛЕКТРИЧЕСКИМ ТОКОМ 	Этот знак может находиться снаружи или внутри оборудования, например, привода или электродвигателя, с целью предупреждения людей о возможном присутствии опасного уровня напряжения.
ОПАСНОСТЬ ОЖОГА 	Этот знак может находиться снаружи или внутри оборудования, например, привода или электродвигателя, с целью предупреждения людей об опасных температурных значениях.

Allen-Bradley, Rockwell Automation, TechConnect, CompactLogix, POINT I/O, FLEX I/O, PowerFlex 70, PowerFlex 40, RSLogix 5000, KwikLink, RSNetWorx, RSNetWorx для ControlNet, RSNetWorx для DeviceNet, ControlFlash, RSLinx, RSView, RSView Studio, Compact I/O, NetLinx, FactoryTalk Activation, RSView Machine Edition, RSViewME, RSLinx Classic и PanelView Plus являются торговыми знаками компании Rockwell Automation.

Торговые знаки, не принадлежащие компании Rockwell Automation, являются собственностью соответствующих компаний.

Предисловие	Назначение данного руководства	5
	Как использовать данное руководство	5
	Для заметок:	6
	Глава 1	
Программирование на языке функциональных блок-схем	Введение	7
	Выбор элементов функционального блока	8
	Выбор имени тега для элемента	8
	Определение порядка выполнения	10
	Фиксация данных	10
	Порядок выполнения	12
	Организация циклов	13
	Разрешение потока данных между двумя блоками	15
	Создание задержки на одно сканирование	15
	Резюме	16
	Определение коннекторов	16
	Задание режима управления (программный/оператор)	17
	Добавление листа	20
	Добавление элемента на языке функциональных блоков	20
	Создание текстового окна	21
	Переключение языка документации	22
	Соединение элементов	23
	Показать или скрыть контакт	23
	Связать элементы между собой	24
	Пометить связь указателем Assume Data Available	24
	Присвоение нового тега	25
	Создание и присвоение нового тега	25
	Присвоить существующий тег	26
	Присваивание непосредственного значения (постоянной)	26
	Использование IREF	26
	Ввод значения в тег блока	27
	Соединение блоков с OCON и ICON	27
	Добавление OCON	27
	Добавление ICON	28
	Проверка процедуры	28
	Служба поддержки Rockwell Automation	34
	Содействие при установке	34
	Возврат продукции	34

Назначение данного руководства

В данном руководстве рассматривается программирование контроллеров Logix5000 при помощи языка программирования функциональных блок-схем (ФБС). Данное руководство входит в состав комплекта руководств на эту тему, в которых рассматриваются общие методики программирования и использования контроллеров Logix5000. Полный перечень руководств по общим методикам приведен в руководстве "Общая методика для программируемых контроллеров Logix5000, публикация 1756-PM001.

Под понятием контроллер Logix5000 имеется в виду любой контроллер на базе операционной системы Logix5000, например:

- Контроллеры CompactLogix
- Контроллеры ControlLogix
- Контроллеры DriveLogix
- Контроллеры FlexLogix
- Контроллеры SoftLogix

Как использовать данное руководство

Форматирование некоторых частей текста отличается от форматирования остального текста.

Форматирование текста	Обозначает	Например	Означает
<i>курсив</i>	Название элемента, отображаемое на экране или в примере	Щелкните правой кнопкой мыши на <i>User-Defined...</i>	Щелкните правой кнопкой мыши по элементу с названием User-Defined.
шрифт courier	данные, которые вы должны подставить в соответствии с вашим приложением (переменная)	Щелкните правой кнопкой мыши на <i>name_of_program...</i>	Вы должны выбрать конкретную программу в вашем приложении. Обычно так обозначаются определяемые пользователем переменные.
в скобках	клавиша на клавиатуре	Нажмите [Enter].	Нажмите клавишу Enter.

Для заметок:

Программирование на языке функциональных блок-схем

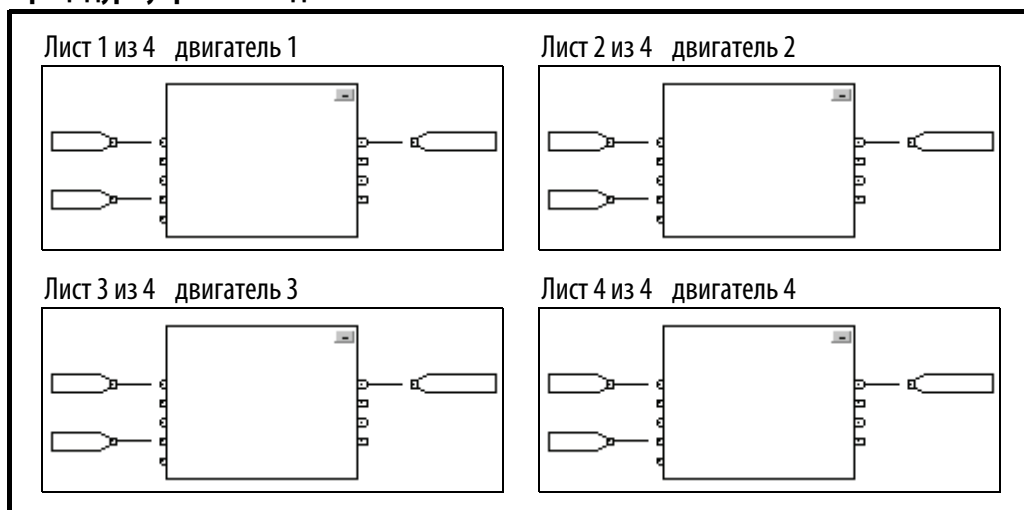
Введение

Для упрощения перемещения по процедуре на языке функциональных блоков разделите процедуру на несколько листов.

- Использование листов позволит вам организовать размещение функциональных блоков и облегчить их поиск. Расположение на листах не влияет на порядок выполнения функциональных блоков.
- При выполнении процедуры выполняются все листы.
- Общей рекомендацией является использование одного листа для одного прибора (двигателя, клапана и т.д.).

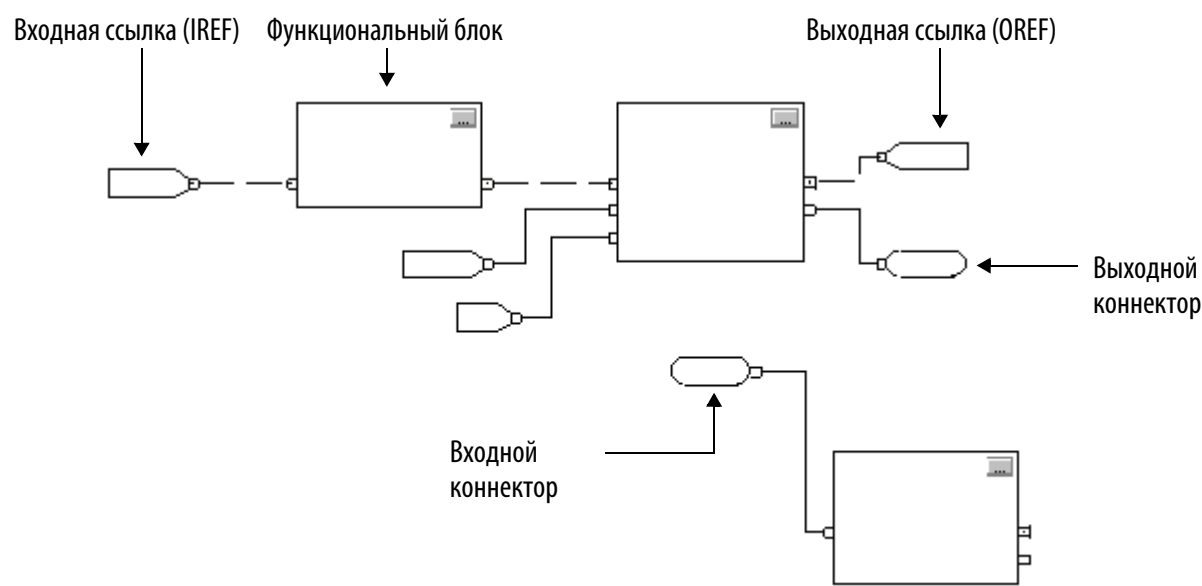
ПРИМЕР

Процедура управления двигателем



Выбор элементов функционального блока

Для управления устройством используйте следующие элементы:



Для выбора элементов функционального блока:

Если вы хотите:	Используйте:
Получить величину из устройства ввода или тега	Входная ссылка (Input reference (IREF))
Отправить величину в выходное устройство или тег	Выходная ссылка (output reference (OREF))
Выполнить операцию над входной величиной или величинами и выдать выходную величину или величины	Функциональный блок
Осуществляйте передачу данных между функциональными блоками при помощи коннекторов, если они: <ul style="list-style-type: none">• на большом расстоянии друг от друга на одном листе.• на разных листах в пределах одной процедуры.	Выходной коннектор связи (output wire connector (OCON) и входной коннектор связи (input wire connector (ICON))
Распределить данные по нескольким точкам в одной процедуре	Один выходной коннектор связи (OCON) и несколько входных коннекторов связи (ICON)

Выбор имени тега для элемента

Каждый функциональный блок использует тег для хранения конфигурации и информации о состоянии для данной инструкции.

- Когда вы добавляете инструкцию на языке функциональных блоков, программное обеспечение RSLogix 5000 автоматически создает для этого блока тег. Вы можете использовать этот тег как он есть, переименовать этот тег или присвоить другой тег

- Для IREF и OREF вы можете создать тег или присвоить существующий.

Для:	Задайте:
Тега	<i>tag_name</i>
Номера бита в составе тега целочисленного типа	<i>tag_name.bit_number</i>
Элемента структуры	<i>tag_name.member_name</i>
Элемента одномерного массива	<i>tag_name[x]</i>
Элемента двумерного массива	<i>tag_name[x,y]</i>
Элемента трехмерного массива	<i>tag_name[x,y,z]</i>
Элемента массива в пределах структуры	<i>tag_name.member_name[x]</i>
Члена элемента массива	<i>tag_name[x,y,z].member_name</i>

Где:

x - позиция элемента для первой размерности,

y - позиция элемента для второй размерности,

z - позиция элемента для третьей размерности.

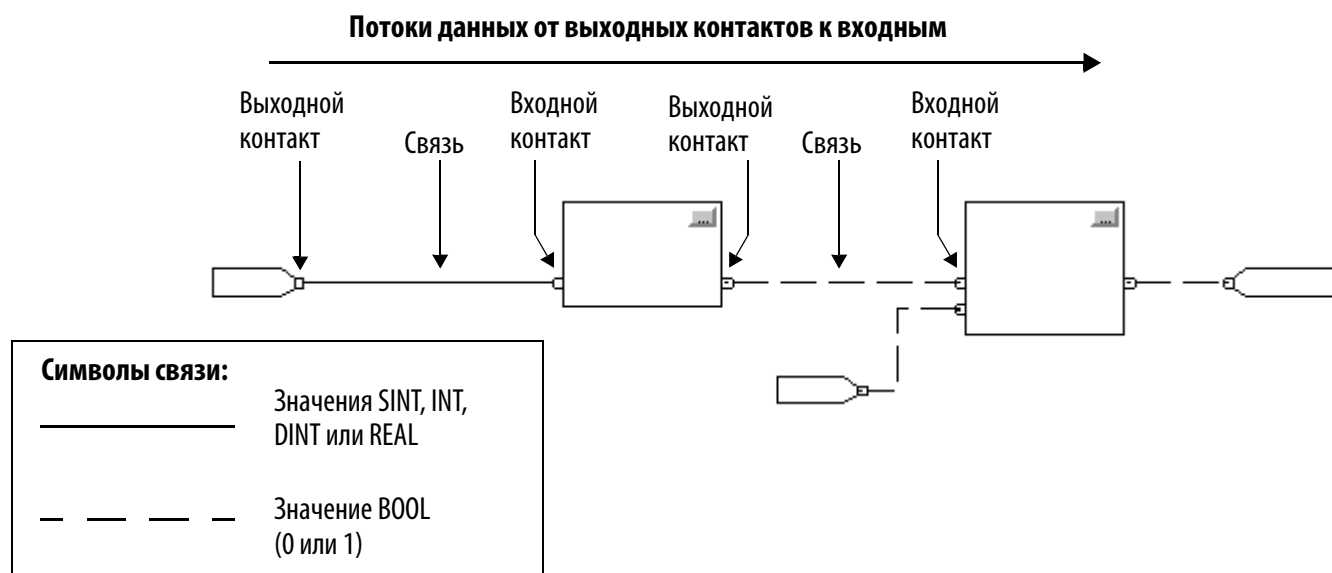
Для структуры в структуре дополнительно добавьте *.member_name*.

СОВЕТ

Данные модуля ввода/вывода обновляются не синхронно с выполнением логики. Если в вашей логике вы ссылаетесь на вход несколько раз, этот вход может менять состояние между отдельными ссылками. Если вы хотите, чтобы вход был в одном и том же состоянии для каждой ссылки, поместите значение входа в буфер и создавайте ссылку на этот тег в буфере.

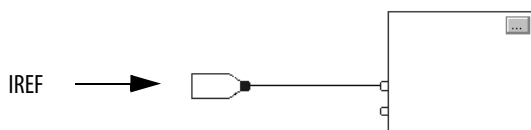
Определение порядка выполнения

Вы определяете порядок выполнения (поток данных), связывая функциональные блоки между собой и указывая, если это необходимо обратные связи. Месторасположение функциональных блоков не влияет на порядок, в соответствии с которым они выполняются.

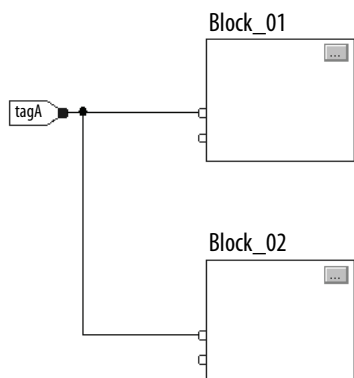


Фиксация данных

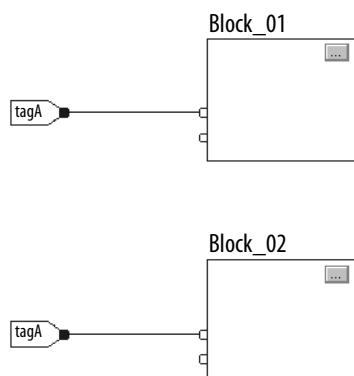
Если вы используете элемент IREF для того, чтобы задать исходные данные для какого-нибудь функционального блока, данные в этом элементе «защелкиваются» для сканирования подпрограммы этого функционального блока. Элемент IREF «защелкивает» данные для тегов программы и контроллера. Контроллер обновляет все данные в IREF в начале каждого сканирования.



В данном примере значение тега tagA сохраняется в начале выполнения процедуры. Это сохраненное значение используется при выполнении блока Block_01. Это же самое значение используется при выполнении блока Block_02. Если значение tagA изменяется во время выполнения процедуры, значение tagA в элементе IREF не изменяется до следующего выполнения процедуры.

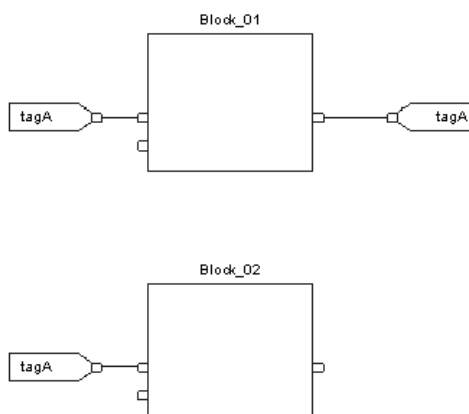


Этот пример аналогичен представленному выше. Значение tagA сохраняется один раз в начале выполнения процедуры. Процедура использует это значение внутри себя.



Начиная с 11 версии программного обеспечения RSLogix 5000, вы можете использовать один и тот же тег в нескольких элементах IREF и одном элементе OREF в одной процедуре. Поскольку значения тегов в элементах IREF «защелкиваются» при каждом сканировании процедуры, все IREF будут использовать одно и то же значение, даже если OREF будет получать другие значения тегов при выполнении этой процедуры. В данном примере, если tagA имеет значение 25,4, то, когда эта процедура начинает выполняться, это значение сканируется и Block_01 изменяет значение tagA на 50,9, второй элемент IREF подключенный к Block_02 будет по-прежнему использовать значение 25,4 при выполнении Block_02 на этом сканировании. Новое значение tagA равное 50,9 не будет

использоваться никаким элементом IREF в данной программе, пока не начнется следующее сканирование.



Порядок выполнения

Программное обеспечение RSLogix 5000 автоматически определяет порядок выполнения функциональных блоков в программе, когда вы:

- проверяете процедуру на языке функциональных блоков,
- проверяете проект, содержащий процедуру на языке функциональных блоков,
- загружаете проект, содержащий процедуру на языке функциональных блоков.

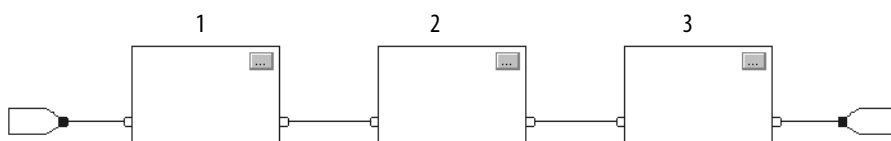
Вы определяете порядок выполнения, связывая функциональные блоки между собой и указывая, если это необходимо, поток данных обратных связей.

Если функциональные блоки не связаны друг с другом, то не имеет значения, какой блок выполняется первым. Между этими блоками нет потока данных.

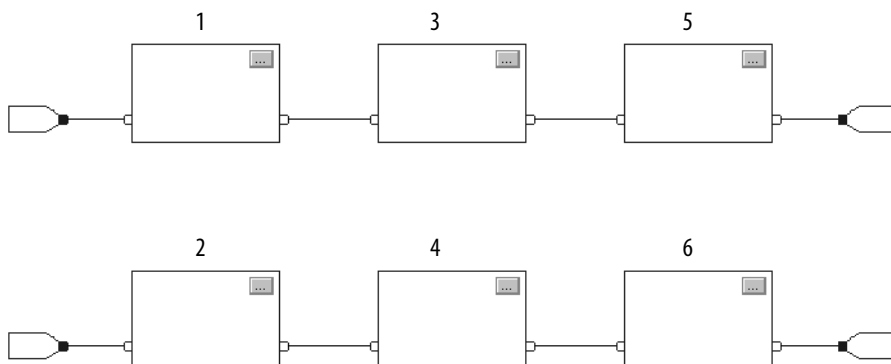


Если вы связали блоки последовательно, порядок выполнения идет от входа к выходу. Входы блока требуют наличия данных до того, как этот блок начнет выполняться контроллером. Например, block 2

должен выполняться раньше блока block 3, потому что block 2 предоставляет входные данные для блока block 3.



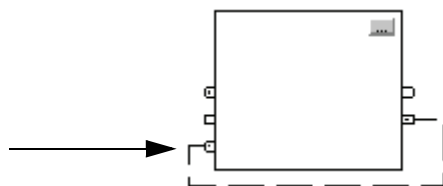
Порядок выполнения блоков имеет значение только для блоков, связанных друг с другом. Следующий пример справедлив, поскольку две группы блоков не связаны друг с другом. Блоки в пределах заданной группы выполняются в последовательности, определяемой их положением внутри этой группы.



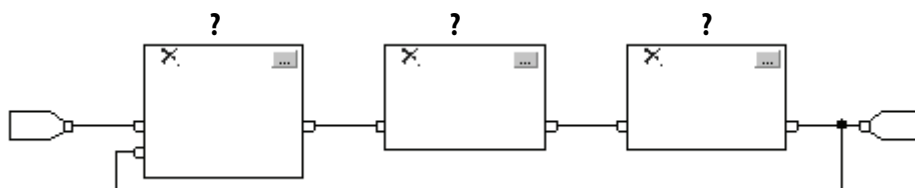
Организация циклов

Для создания цикла обратной связи свяжите выходной контакт блока с входным контактом того же блока. Ниже приведен пример. Цикл содержит один единственный блок и порядок выполнения роли не играет.

Представленный контакт входа использует выходные данные, произведенные блоком во время предыдущего сканирования.

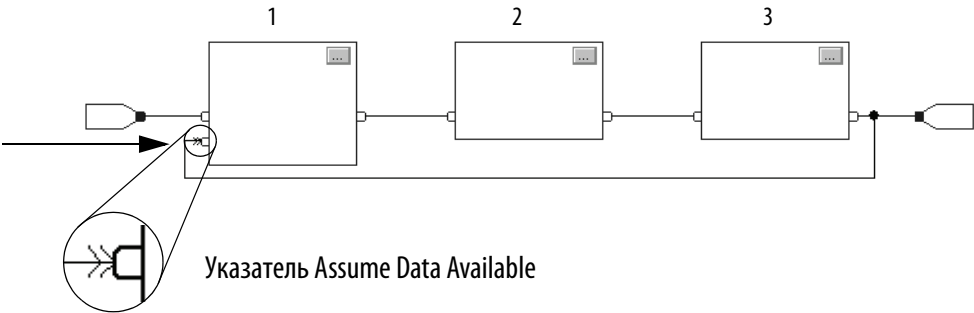


Если группа блоков объединена в цикл, то контроллер не может определить какой блок выполнять первым. Другими словами, он не сможет разрешить цикл.



Чтобы определить блок, который будет выполняться первым, пометьте входной контакт, который создает цикл (обратная связь), при помощи указателя Assume Data Available (допустим, что данные имеются). В следующем примере блок block 1 использует результат блока 3, который был получен при предыдущем выполнении программы.

Этот входной контакт использует выход, произведенный блоком 3 при предыдущем сканировании.



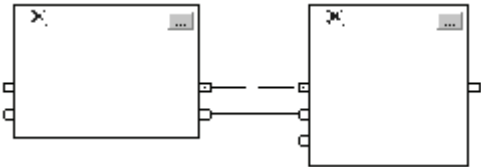
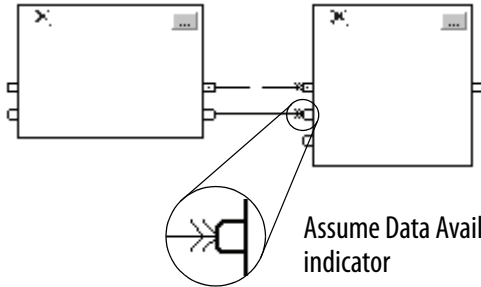
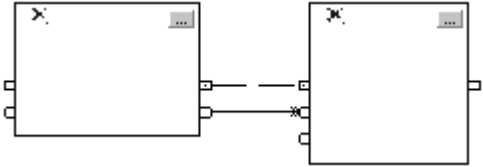
Указатель Assume Data Available определяет поток данных внутри цикла. Стрелка указывает, что эти данные служат входом для первого блока цикла.

Не помечайте все связи цикла указателем Assume Data Available.

Правильный вариант	Неправильный вариант
<div></div> <p>Указатель Assume Data Available определяет поток данных внутри цикла.</p>	<div></div> <p>Контроллер не может разрешить цикл, поскольку для всех связей используется указатель Assume Data Available.</p>

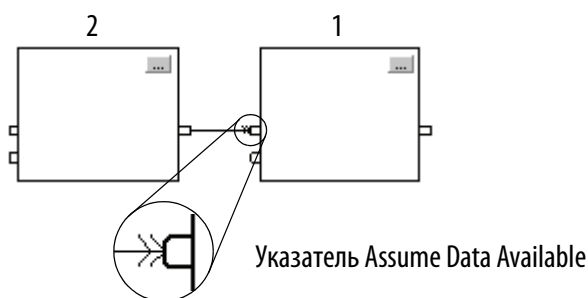
Разрешение потока данных между двумя блоками

Если вы используете две или более связи между двумя блоками, используйте одинаковые указатели потока данных для всех связей между двумя блоками.

Правильный вариант	Неправильный вариант
 <p>Ни та, ни другая связь не используют указатель Assume Data Available.</p>  <p>Assume Data Available indicator</p> <p>Обе связи используют указатель Assume Data Available.</p>	 <p>Одна связь использует указатель Assume Data Available, в то время как другая нет.</p>

Создание задержки на одно сканирование

Для того чтобы организовать задержку на одно сканирование между блоками, используйте указатель Assume Data Available. В следующем ниже примере блок block 1 выполняется первым. Он использует выход блока, который получается на предыдущем сканировании процедуры.



Резюме

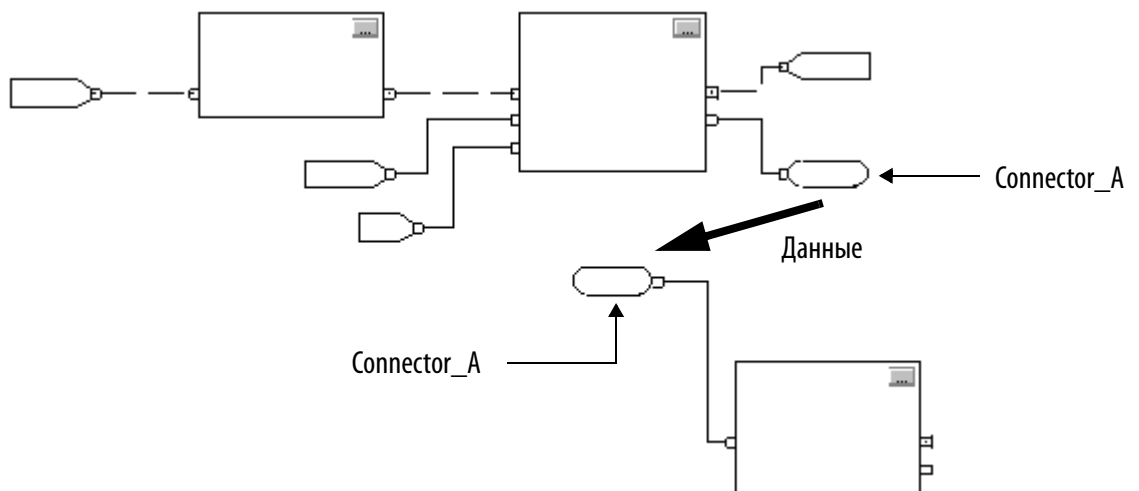
В итоге, процедура на языке функциональных блоков выполняется следующим образом:

1. Контроллер «защелкивает» все данные в элементах IREF.
2. Контроллер выполняет функциональные блоки в той последовательности, в которой они связаны.
3. Контроллер записывает результаты в элементы OREF.

Определение коннекторов

Как и связи, коннекторы передают потоки данных от выходных контактов к входным контактам. Используйте коннекторы, когда:

- элементы, которые вы хотите соединить, расположены на разных листах в одной процедуре,
- Размещение на релейной схеме связи затруднено ввиду наличия других связей или элементов.
- вы хотите распределить данные по нескольким точкам в процедуре.



При использовании коннекторов следуйте следующим правилам:

- Каждый OCON требует уникального имени.
- Каждому OCON должен соответствовать, по крайней мере, один ICON (например, ICON с таким же именем, как и OCON).
- Несколько ICON могут ссылаться на один OCON. Это позволяет вам распределять данные по нескольким точкам в процедуре.

Задание режима управления (программный/оператор)

Несколько инструкций поддерживают режимы управления программный/оператор. Эти инструкции:

- Улучшенный выбор (Enhanced Select (ESEL))
- Сумматор (Totalizer (TOT))
- Улучшенный ПИД (PIDE)
- Линейное измерение/выдержка (RMPS)
- Дискретное 2х режимное устройство (D2SD)
- Дискретное 3-х режимное устройство (D3SD)

Способ управления программа/оператор позволяет вам управлять этими инструкциями одновременно как из пользовательской программы, так и при помощи интерфейса оператора. При программном управлении (Program) инструкция управляется при помощи программных входов (Program inputs) для этой инструкции, а при управлении оператором (Operator), инструкция управляется входами оператора (Operator inputs).

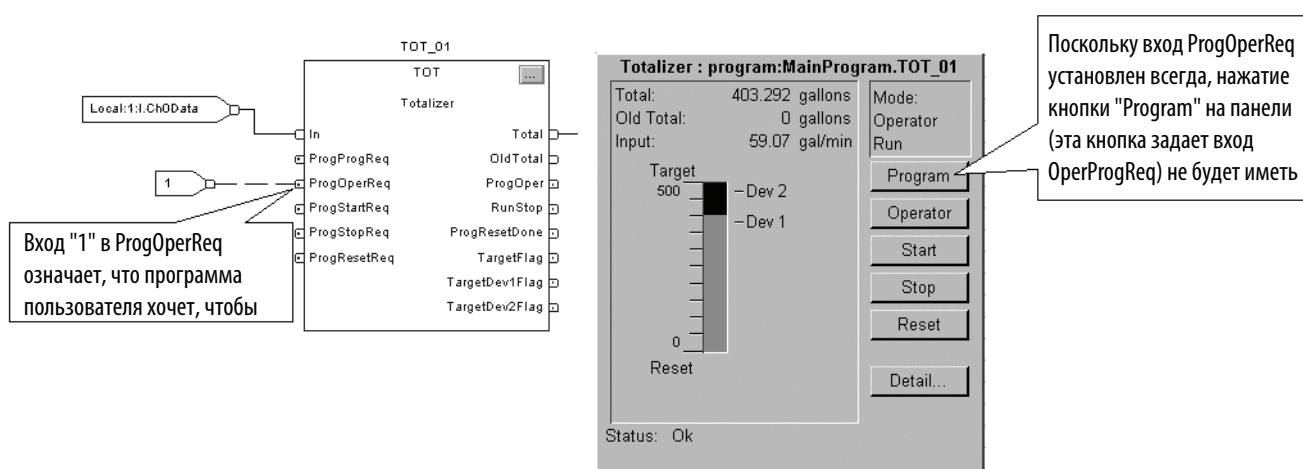
Режимы управления Program или Operator задаются при помощи следующих входов:

Вход:	Описание:
.ProgProgReq	Программный запрос на переход к программному управлению (Program).
.ProgOperReq	Программный запрос на переход к управлению оператором (Operator).
.OperProgReq	Запрос оператора на переход к программному управлению.
.OperOperReq	Запрос оператора на переход к управлению оператором.

Для того чтобы определить, в каком режиме управления находится инструкция (Program или Control), проверьте выходное значение ProgOper. Если ProgOper установлен, инструкция находится в режиме управления Program; если ProgOper отсутствует, инструкция находится в режиме управления Operator.

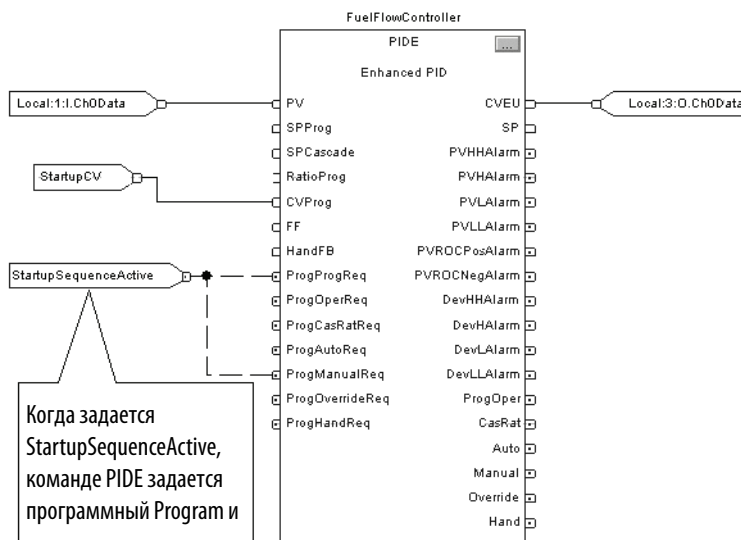
Если установлены оба бита запроса режима управления, то режим Operator имеет приоритет над режимом Program. Например, если оба бита ProgProgReq и ProgOperReq установлены, инструкция переходит в режим управления Operator.

Входы программных запросов (Program request) имеют приоритет над входами запросов оператора (Operator request). Это дает возможность использовать входные значения ProgProgReq и ProgOperReq для «фиксации» инструкции в желаемом режиме управления. Например, допустим, что инструкция Totalizer будет всегда работать под управлением оператора, и ваша программа никогда не сможет управлять запуском или остановом инструкции Totalizer. В этом случае вы должны ввести (подсоединить) литерную величину 1 в ProgOperReq. Это не даст возможности оператору перевести инструкцию Totalizer в программный режим управления, задав OperProgReq с интерфейса оператора.



Аналогично, постоянная установка ProgProgReq может «зафиксировать» инструкцию в программном режиме управления. Это полезно при последовательном включении, когда вы хотите, чтобы программа управляла работой инструкции не опасаясь, что оператор, по невнимательности, возьмет управление над инструкцией. В этом примере программа задает вход ProgProgReq при запуске, а затем очищает ProgProgReq, когда запуск завершен. После того как ProgProgReq сброшен, инструкция останется в программном режиме управления, пока не получит запрос на его изменение. Например, оператор может задать OperOperReq с панели для того, чтобы взять управление инструкцией в свои руки.

В следующем примере показано как «зафиксировать» инструкцию в программном режиме управления.



Входные значения запроса оператора в инструкцию всегда сбрасываются инструкцией при выполнении. Это позволяет интерфейсам оператора работать с этими инструкциями, просто задавая бит запроса желаемого режима. Вам не нужно программировать интерфейс оператора на сброс битов запроса. Например, если интерфейс оператора задает вход OperAutoReq для инструкции PIDE, то когда выполняется инструкция PIDE, она определяет, каким должен быть соответствующий отклик, и сбрасывает OperAutoReq.

Входные значения программных запросов обычно не сбрасываются инструкцией, поскольку они задаются как входные связи. Если инструкция сбросит эти входные значения, они будут восстановлены элементом входной связи. Может возникнуть ситуация, когда вы захотите задать программные запросы (Program requests) таким образом, чтобы они сбрасывались инструкцией. В этом случае вы можете задать вход ProgValueReset, и программа будет всегда сбрасывать значения программных запросов при выполнении.

В этом примере цепочка релейной логики в другой процедуре используется для разового «защелкивания» ProgAutoReq для инструкции PIDE при нажатии на пусковую кнопку. Поскольку инструкция PIDE автоматически сбрасывает программные запросы, у вас нет необходимости записывать какой либо алгоритм релейных схем для того, чтобы сбрасывать ProgAutoReq после выполнения

процедуры, и инструкция будет получать только один запрос на переход в режим Auto при каждом нажатии кнопки.

Когда TIC101AutoReq Pushbutton нажата, происходит разовое защелкивание ProgAutoReq для инструкции PIDE TIC101. TIC101 уже была сконфигурирована с заданным входным значением ProgValueReset так, что когда инструкция PIDE выполняется, она автоматически сбрасывает ProgAutoReq.



Добавление листа

Чтобы добавить лист к процедуре на языке функциональных блоков сделайте следующее:

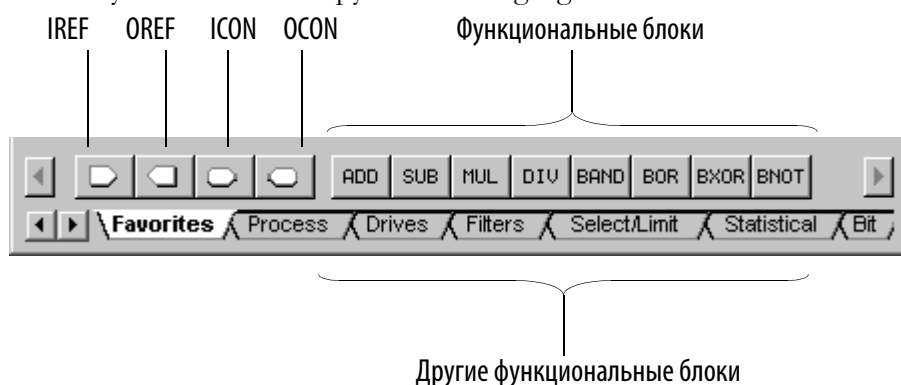
1. Щелкните на 



2. Введите описание листа (до 50 символов).

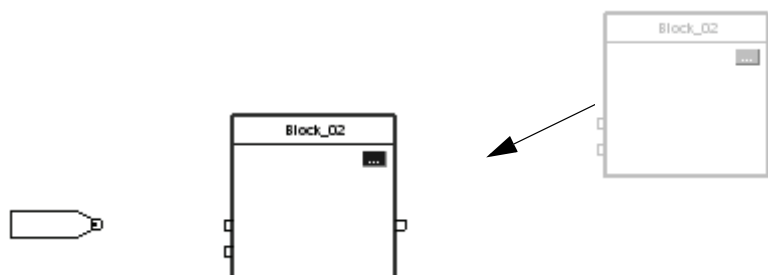
Добавление элемента на языке функциональных блоков

Для добавления элемента на языке функциональных блоков используйте панель инструментов Language Element.

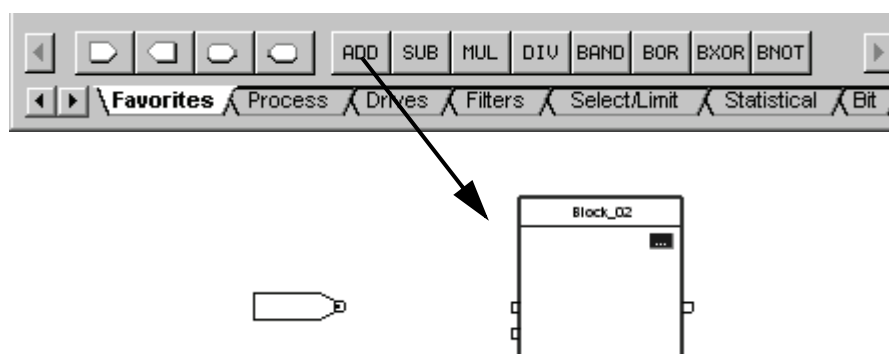


Чтобы добавить элемент:

1. На панели инструментов Language Element (Элементы языка) щелкните на кнопке того элемента, который вы хотите добавить.
2. Перетащите этот элемент в нужное место.



Вы можете так же перетащить кнопку донного элемента в нужное место.



Создание текстового окна

При помощи текстового окна вы можете добавлять примечания, поясняющие функции элемента ФБС, например, блока, ссылки и коннектора. Текстовые окна сохраняются только в автономном файле проекта ACD. Текстовые окна не загружаются в память контроллера.

Вы также можете использовать текстовое окно для фиксирования информации, которая может понадобиться вам позднее. Пример приведен ниже.



1. Щелкните на

Возникает текстовое окно.



2. Перетащите текстовое окно к элементу, к которому оно относится.
3. Дважды щелкните на текстовом окне и введите примечание.
4. Затем нажмите [Ctrl] + [Enter].
5. Что должно произойти с текстовым окном при удалении элемента ФБС?

Если Вы хотите, чтобы текстовое окно:	То:
Оставалось на том же месте	Больше действий не требуется.
Перемещалось вместе с элементом, к которому оно относится	Перейдите к пункту 6.



Зеленая точка

6. Щелкните по значку контакта в текстовом окне, затем щелкните на элементе, к которому вы хотите прикрепить окно.

Зеленая точка указывает подходящую точку связи.

Переключение языка документации

В 17 версии программного обеспечения RSLogix 5000 появилась возможность отображения документации по проекту, такой как описания тегов и комментарии по звеньям на любом поддерживаемом языке перевода. Вы можете хранить документацию на различных языках в одном файле проекта, а не в отдельных файлах проекта для конкретных языков. Вы задаёте все языки перевода, которые будет поддерживать проект, а также устанавливаете текущий язык, язык по умолчанию и пользовательский язык перевода. Программное обеспечение будет использовать язык по умолчанию для отдельного компонента проекта там, где будет отсутствовать текст на текущем языке. Однако

вы можете использовать пользовательский язык для изменения документации для различных пользователей.

Добавьте переведенные описания в ваш проект RSLogix 5000 либо при программировании на этом языке, либо с использованием средства импорта/экспорта для перевода документации в автономном режиме и ее последующего импорта обратно в проект. Как только вы активируете переключение языков в программном обеспечении RSLogix 5000, вы сможете динамически переключать языки во время использования этого программного обеспечения.

В рамках проекта возможность использования нескольких языков поддерживается для следующей документации:

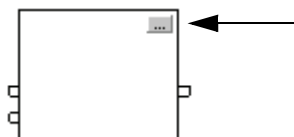
- Описания компонентов в тегах, процедурах, программах, пользовательских типах данных и дополнительных инструкциях.
- Фазы оборудования.
- Тренды.
- Контроллеры.
- Сообщения о тревогах (для конфигураций ALARM_ANALOG и ALARM_DIGITAL).
- Задачи.
- Описания свойств модулей в средстве обзора контроллера.
- Комментарии по звеньям, текстовые окна ПФС и ФБС.

Более подробная информация по обеспечению поддержки нескольких переводов документации в проекте приведена в интерактивном справочнике.

Соединение элементов

Показать или скрыть контакт

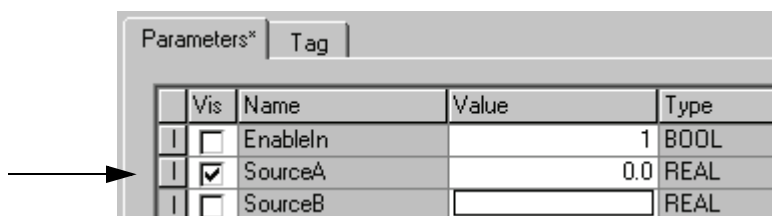
Когда вы добавляете инструкцию на языке функциональных блоков, блок возникает как набор контактов для параметров по умолчанию. Остальные контакты скрыты. Чтобы показать или скрыть контакт:



1. Щелкните на [...] кнопку блока.

2. Установите или снимите флажки напротив контактов:

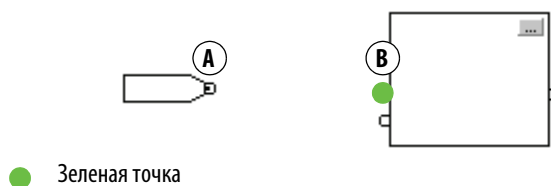
Если вы хотите:	То:
Скрыть контакт	Снимите флажок напротив.
Показать контакт	Установите флажок напротив.



3. Щелкните на ОК.

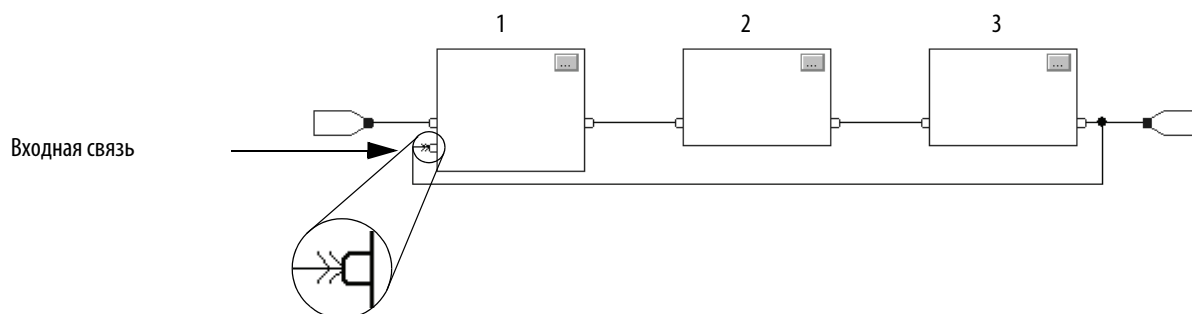
Связать элементы между собой

Чтобы связать два элемента между собой, щелкните на выходном контакте первого элемента, а затем щелкните на входном контакте второго элемента. Зеленая точка указывает подходящую точку связи.



Пометить связь указателем Assume Data Available

Чтобы пометить связь как входную, щелкните правой клавишей мыши и выберите Assume Data Available.



Присвоение нового тега



Создание и присвоение нового тега

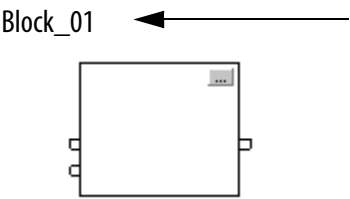
1. Дважды щелкните в области операнда.
2. Введите имя тега и нажмите [Enter].
3. Правой кнопкой мыши щелкните на имени тега и выберите New “tag_name” (Новое «имя_тега»).

4. Щелкните на ...

5. Выберите для этого тега тип данных.
6. Если вы хотите определить тег как массив, введите число элементов для каждой размерности.
7. Щелкните на ОК.
8. Выберите для тега область применимости.

9. Щелкните на ОК.

Присвоить существующий тег



- 1. Дважды щелкните в области операнда.
- 2. Щелкните на ▼.
- 3. Выберите тег:

Для выбора:	Сделайте следующее:
Тега	Дважды нажмите на название тега.
Битового номера	А. Нажмите на название тега. В. Справа от названия тега нажмите на ▼. С. Выберите требуемый бит.

- 4. Нажмите [Enter] или щелкните в другом месте схемы.

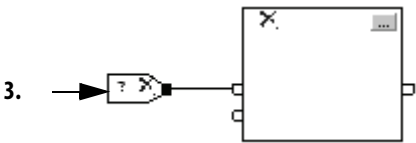
Присваивание
непосредственного значения
(постоянной)

Для того чтобы входному параметру присвоить постоянное значение вместо значения тега, вы можете использовать следующие опции:

Если вы хотите:	То:
Сделать значение видимым на схеме или отчетах	Используйте IREF
Иметь возможность изменения значения через подключение без редактирования процедуры	Введите значение в тег блока

Использование IREF

- 1. Добавьте IREF.
- 2. Свяжите IREF с входным контактом, который получает данное значение.

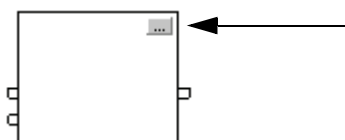


3. Дважды щелкните в области операнда IREF.

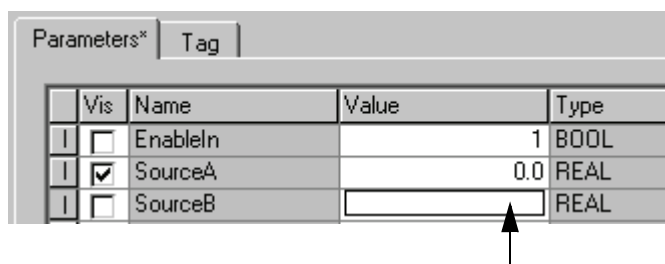
4. Введите значение и нажмите [Enter].

Ввод значения в тег блока

Чтобы присвоить значение параметру, когда связь подключается к контакту:



1. Щелкните на кнопку блока.

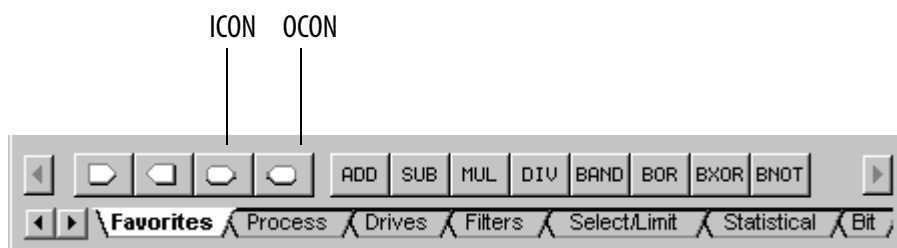


2. Введите значение.

3. Щелкните на ОК.

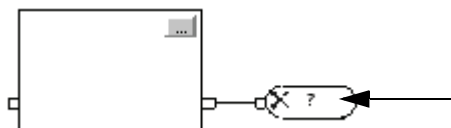
Соединение блоков с OCON и ICON

Используйте OCON или ICON для передачи данных между листами или при комплексной связи:



Добавление OCON

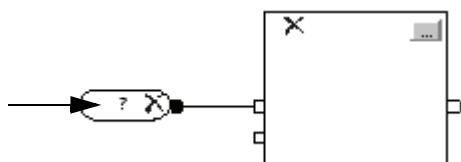
1. Добавьте коннектор выходной связи (OCON) и поместите его около выходного контакта, который предоставляет данное значение.



2. Соедините OCON с этим выходным контактом.
3. Дважды щелкните в области операнда этого OCON.
4. Введите имя, которое идентифицирует этот коннектор, и нажмите [Enter].

Добавление ICON


1. Добавьте коннектор входной связи (ICON) и поместите его около входного контакта, который получает значение от соответствующего OCON.



2. Соедините ICON с входным контактом.
3. Дважды щелкните в области операнда этого ICON.
4. Выберите имя OCON, передающего значение этому коннектору и затем щелкните на пустой области данной схемы.

Проверка процедуры

По мере того, как вы программируете процедуру, периодически проверяйте свою работу:

1. В самой верхней линейке инструментов окна RSLogix 5000 щелкните на .
2. Если внизу окна будут перечислены какие-либо ошибки:
 - а. Перейдите на первую ошибку или предупреждение и нажмите [F4].
 - б. Исправьте ошибку в соответствии с описанием в окне Results (Результаты).
 - в. Вернитесь к пункту 1.
3. Чтобы закрыть окно Results (Результаты), нажмите [Alt] + [1].

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.

Pub. Title/Type	CompactLogix SystemLogix5000 Controllers Function Block Diagram						
Cat. No.	1769-L31, 1769-L32C, 1769-L32E, 1769-L35CR, 1769-L35E1756 ControlLogix, 1769 CompactLogix, 1789 SoftLogix, 1794 FlexLogix, PowerFlex 700S with DriveLogix	Pub. No.	1756-PM009B-RU-P	Pub. Date	July 2008	Part No.	957955-62

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

Overall Usefulness	1	2	3	How can we make this publication more useful for you?		
Completeness (all necessary information is provided)	1	2	3	Can we add more information to help you?		
				procedure/step	illustration	feature
				example	guideline	other
				explanation	definition	
Technical Accuracy (all provided information is correct)	1	2	3	Can we be more accurate?		
				text	illustration	
Clarity (all provided information is easy to understand)	1	2	3	How can we make things clearer?		
Other Comments	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

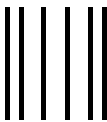
Return this form to: Rockwell Automation Technical Communications, 1 Allen-Bradley Dr., Mayfield Hts., OH 44124-9705

Fax: 440-646-3525 Email: RADocumentComments@ra.rockwell.com

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



Служба поддержки Rockwell Automation

Компания Rockwell Automation предоставляет техническую информацию в интернете с целью поддержки своих клиентов. По адресу <http://support.rockwellautomation.com> вы найдете технические руководства, ответы на часто задаваемые вопросы, заметки по техническим характеристикам и эксплуатации продукции, коды пользования пакетами ПО для ознакомления и загрузки, а также службу поддержки MySupport, которую Вы можете настроить по своему желанию с целью оптимального использования перечисленных средств клиентской поддержки.

Наша компания также предлагает своим клиентам программы поддержки пользователей по телефону TechConnect по вопросам установки, настройки и разрешения проблем. Более подробные сведения вы можете получить у дистрибьюторов и представителей компании Rockwell Automation в вашем регионе или на сайте <http://support.rockwellautomation.com>

Содействие при установке

Если у вас возникли проблемы в течении первых 24 часов процесса установки, пожалуйста, обратитесь к информации, содержащейся в настоящем руководстве. Вы также можете позвонить по специальному телефону Службы поддержки клиентов и проконсультироваться по вопросам приобретения и эксплуатации продукции нашей компании.

Соединенные Штаты	1.440.646.3434 Понедельник – Пятница, 8.00 – 17.00 (Восточное стандартное время США)
За пределами США	Пожалуйста, обратитесь к представителю компании Rockwell Automation в вашем регионе по любым интересующим вас вопросам.

Возврат продукции

Компания Rockwell Automation проводит испытания всей своей продукции в целях обеспечения контроля качества и пригодности изделий к использованию при отгрузке продукции с производства. Тем не менее, в случае если изделие не функционирует и подлежит возврату, выполните следующие шаги.

Соединенные Штаты	Для осуществления процесса возврата вы должны предоставить ваш личный идентификационный номер в службе поддержки клиентов (узнать его можно, позвонив по вышеуказанному телефону) дистрибьютору компании в вашем регионе.
За пределами США	Пожалуйста, обратитесь к представителю компании Rockwell Automation в вашем регионе по вопросу возврата изделия.

www.rockwellautomation.com

Штаб-квартира по решениям в энергетике, управлении и информации

Россия и СНГ: Rockwell Automation BV, 115054, Москва, Большой Строченовский пер., 22/25, офис 402, Тел. +7(495)956-0464, факс +7(495)956-0469

Америка: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204 USA, тел.: (1) 414 382-2000, факс: (1) 414 382-4444

Европа/Ближний Восток/Африка: Rockwell Automation, Vorstlaan/Boulevard de Souverain 36, 1170 Brussels, Belgium, тел.: (32) 2 663 0600, факс: (32) 2 663 0640

Тихоокеанский регион: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, тел.: (852) 2887 4788, факс (852) 2508 1846