

Редактирование свойств блока и тестирование программы

5

В этой главе

После того как Вы создали и отредактировали блоки данных и логические блоки, Вы должны проверить и отредактировать свойства блока. Они содержат информацию, которая может помочь Вам в идентификации блока и в поиске неисправностей.

Редактор FUP позволяет тестировать отдельный блок при его исполнении в программе пользователя на CPU. Вы можете следить на экране за потоком сигнала внутри сегмента. Этот тест, называемый статусом программы, поможет Вам проверить последовательность работы программы и устранить ошибки.

Обзор главы

В разделе	Вы найдете	на стр.
5.1	Редактирование свойств блока	5–2
5.2	Тестирование программы на FUP. Обзор	5–5
5.3	Настройка окна статуса программы	5–6
5.4	Установка условия запуска	5–7
5.5	Выбор среды тестирования и запуск/остановка статуса программы	5–8

5.1 Редактирование свойств блока

Обзор

В свойствах блока хранится дополнительная информация о блоке. Здесь Вы при необходимости можете задать имя, принадлежность к семейству, версию и автора блока. Свойства включают также другие статистические данные и дополнительную информацию, которая автоматически вносится системой и не может редактироваться пользователем (см. рис. 5–1). Вы можете также назначить системные атрибуты блоку.

Свойства блока дают Вам важную информацию о виде блока, необходимой памяти и времени последнего изменения. Эта информация может оказаться полезной при прослеживании таких проблем, как недостаток памяти и конфликт меток времени.

Последовательность действий

Свойства блока можно редактировать с помощью диалогового окна.

- В SIMATIC Manager выделите блок и выберите команду меню **Edit → Object Properties** [Редактировать → Свойства объекта].
- Если блок уже открыт в редакторе FUP, выберите **File → Properties** [Файл → Свойства] в строке меню.

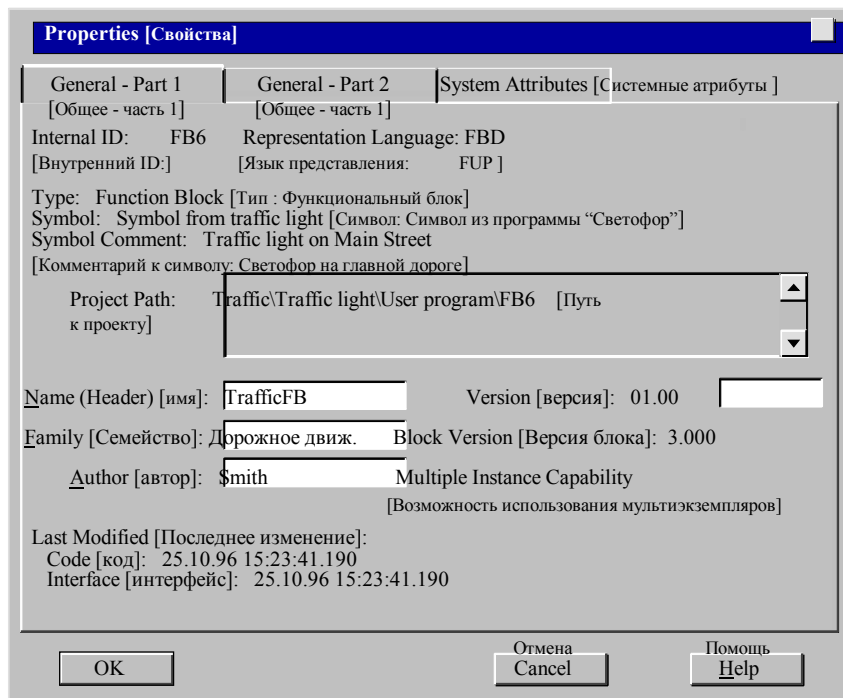
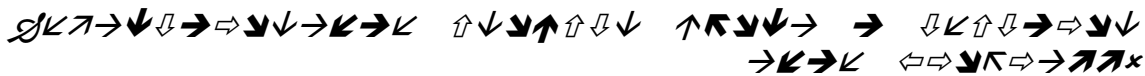


Рис. 5-1. Диалоговое окно “Свойства” для блоков в редакторе FUP

На вкладках ”General – Part 1” [“Общее - часть 1”], ”General – Part 2” [“Общее - часть 2”] и ”System Attributes” [“Системные атрибуты”] Вы можете сделать ряд записей, включая следующие:



Имя и семейство блока

Блоки, для которых указаны имя и семейство, легче классифицировать. Например, Вы можете отнести некоторые блоки к семейству “Системы управления с замкнутым контуром”, показав тем самым, что все они используются для программирования систем управления с замкнутым контуром.

Преимущество этой информации становится очевидным, когда впоследствии блок вызывается в разделе операторов другого блока FUP: семейство и вид этого блока отображается в окне списка “Program Elements” [“Элементы программы”] при выборе блока, так что Вы можете легко определить назначение блока.

Версия блока

Эта информация показывает, какая версия STEP 7 была использована для создания блока. Блоки версии 1 должны быть преобразованы, прежде чем они могут быть включены в программу версии 3. Этого можно достичь в SIMATIC Manager с помощью команды меню **File → Open Version 1 Project** [Файл → Открыть проект версии 1].

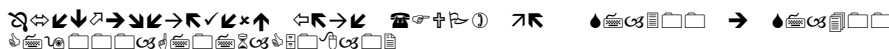
Блоки версии 1 не могут использоваться совместно с мультиэкземплярами. Они должны быть декомпилированы в исходные файлы, а затем скомпилированы в блоки версии 3.

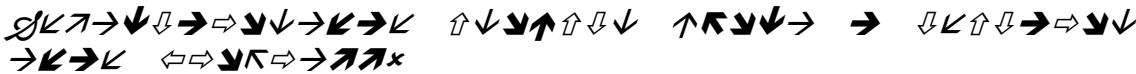
За дополнительной информацией обращайтесь к *Руководству пользователя* /231/.

Атрибуты блока

Атрибуты блока на вкладке “General - Part 2” [“Общее - часть 2”] включают следующие записи:

- Атрибут **“The data block is write-protected in the programmable controller”** [“Блок данных защищен от записи в программируемом контроллере”] означает, что блок предназначен только для чтения. Это полезно для блоков данных, содержащих постоянные значения, которые не должны изменяться. Этот DB должен существовать как исходный файл на AWL.
- Атрибут **“Know-How Protection”** [“Защита ноу-хау”] означает защищенный блок и влияет следующим образом:
 - Раздел операторов нельзя просмотреть.
 - В таблице описания переменных не отображаются временные и статические переменные.
 - Блок не может быть декомпилирован в исходный файл на AWL.
 - Свойства блока не могут редактироваться.
- Атрибут **“Standard Block”** означает защищенный стандартный блок фирмы Siemens. Этот атрибут появляется в нижнем левом углу вкладки.
- Атрибут **“Unlinked”** [“Несвязанный”] появляется только у блоков данных. Он указывает, что блок данных не может быть загружен из загрузочной памяти в рабочую память CPU. К блокам данных в загрузочной памяти можно обратиться только с помощью SFC, которые копируют содержимое блоков данных в рабочую память. Этим достигается более эффективное использование рабочей памяти, так как она содержит во время исполнения программы только нужные данные.





Указание

Такие атрибуты, как защита блока, защита от записи и несвязанность, могут быть добавлены к блоку только в том случае, если он программируется как исходный файл на AWL. Если Вы создали свой блок в FUP, то Вы должны перейти в представление на языке программирования AWL с помощью команды меню **View → STL** [Вид → Список команд]. Затем Вы должны преобразовать блок в исходный файл, прежде чем ввести эти атрибуты. После компиляции исходных файлов в блоки эти блоки будут защищены в соответствии с атрибутами, которые Вы выбрали.
За более подробной информацией обращайтесь к *Справочному руководству по AWL /232/*.

Системные атрибуты для блоков

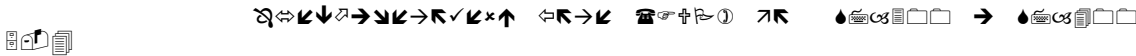
Для конфигурирования управления процессом и диагностики процесса Вы можете назначить следующие атрибута на вкладке "System Attributes" ["Системные атрибуты"].

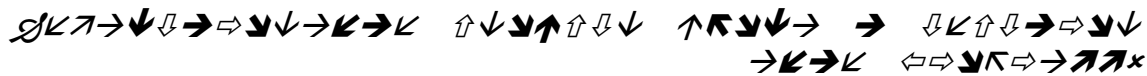
Таблица 5–1. Системные атрибуты для конфигурирования управления процессом

Атрибут	Значение	Когда назначать атрибут	Допустимый тип блока
S7_m_c	true, false [истина, ложь]	Если блок должен управляться или наблюдаться с панели оператора.	FB, SFB
S7_tasklist	'taskname1', 'taskname2', и т.д.	Если блок должен вызываться в организационных блоках, иных, чем циклические OB (например, в OB ошибок или запуска)	FB, SFB, FC, SFC
S7_block-view	big, small [большой, маленький]	Для указания, должен ли блок отображаться в большом или малом формате.	FB, SFB, FC, SFC

Таблица 5–2. Системные атрибуты для диагностики процесса

Атрибут	Значение	Когда назначать атрибут	Допустимый тип блока
S7_pdiag	true, false [истина, ложь]	Если блок должен генерировать информацию, имеющую значение для диагностики процесса.	FB, FC, OB, DB
S7_pdiag_unit	true, false [истина, ложь]	Если блок должен генерировать информацию, имеющую значение для диагностики процесса, и должна наблюдаться единица измерения.	UDT
S7_pdiag_motion	true, false [истина, ложь]	Если блок должен генерировать информацию, имеющую значение для диагностики процесса, и должно наблюдаться движение.	UDT





5.2 Тестирование программы на FUP. Обзор

Метод тестирования

Программу на FUP можно тестировать визуально, отображая поток сигнала внутри сегмента блок. Отображение состояния программы обновляется циклически.

Предварительные условия

Состояние программы можно отобразить только при выполнении следующих условий:

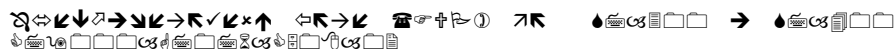
- Блок был сохранен и загружен в CPU без ошибок.
- CPU работает и исполняет программу пользователя.
- Блок открыт в режиме online.

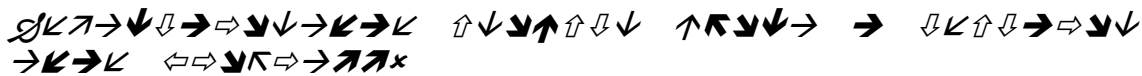
Основная последовательность действий

На рис. 5–2 показана основная последовательность действий для наблюдения статуса программы:



Рис. 5-2. Процедура тестирования логических блоков в FUP





5.3 Настройка окна статуса программы

Последовательность действий

Перед запуском статуса программы в FUP Вы можете выбрать по своему желанию способ отображения потока сигналов, используя команду меню **Options** → **Customize** [Параметры → Настроить] для открытия вкладки "LAD\FBD" (в немецкоязычном варианте "KOP/FUP").

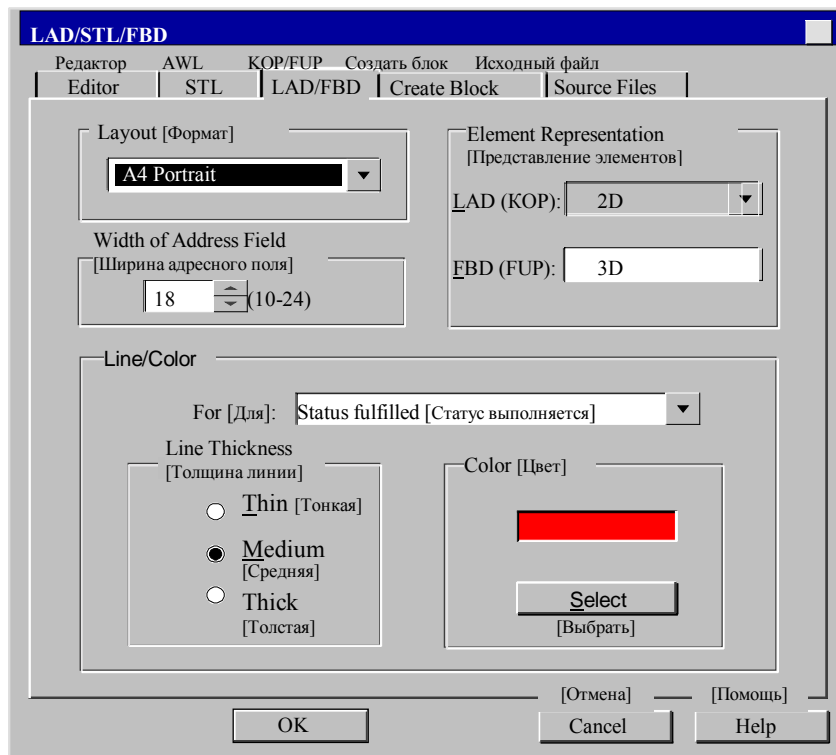
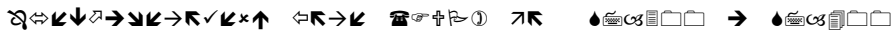
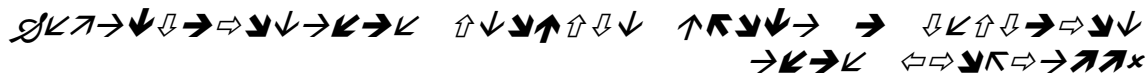


Рис. 5-3. Настройка отображения статуса программы в FUP

На этой вкладке Вы выбираете цвет и толщину линии для двух возможных результатов:

- "Status not fulfilled" ["Статус не выполняется"]: условия не выполнены. Поток сигнала отсутствует (пунктирная линия).
- "Status fulfilled" ["Статус выполняется"]: условия выполнены. Сигнал проходит (сплошная линия).





5.4 Установка условия запуска

Фон

Установкой условий запуска Вы выбираете среду вызова тестируемого блока. Тестирование не производится, пока не выполнено условие запуска.

Последовательность действий

Условия запуска могут быть установлены вызовом команды меню **Debug → Call Environment** [Отладка → Среда вызова].

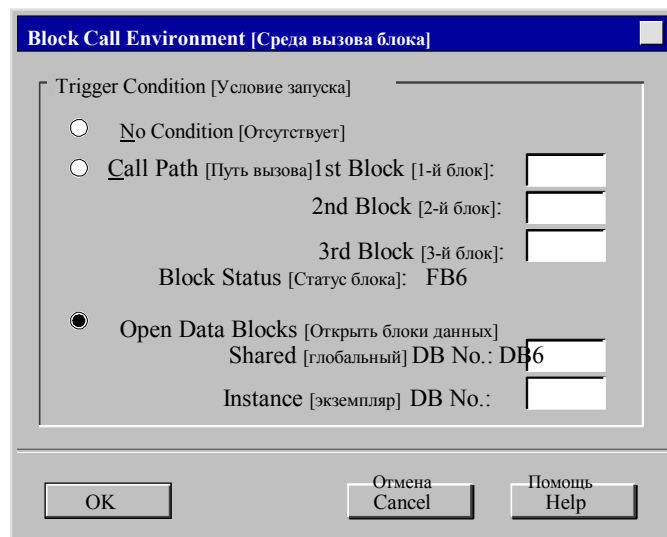
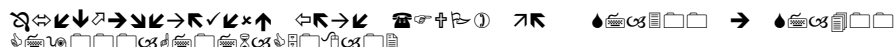


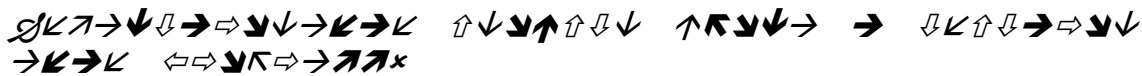
Рис. 5-4. Установка условий запуска

Настройки условий запуска и их значения

Три возможные настройки имеют следующие значения:

- Условия отсутствуют: Среда вызова тестируемого блока не имеет значения. Это значит, что если блок вызывается в процессе выполнения программы из разных точек, то Вы не сможете установить вызов, к которому относится статус.
- Путь вызова: Это путь, используемый для вызова блока, чтоб запустить отображение статуса. Вы можете ввести три уровня вложения блоков, прежде чем будет достигнут блок, подлежащий тестированию.
- Открыть блоки данных: В этом случае Среда вызова определяется одним или двумя блоками данных. Отображение статуса запускается, если тестируемый в данный момент блок вызывается в связке с одним из этих двух блоков данных.





5.5 Выбор среды тестирования и запуск/остановка статуса программы

Выбор среды тестирования

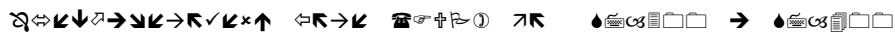
Имеется два способа тестирования программы в режиме online.

- Если установлена среда тестирования ”**process**” [“процесс”], то Ваша программа тестируется в режиме online в работающем процессе. Отображение состояния операторов в программных циклах, выполняемых более одного раза в цикле обработки программы процессором, прерывается в точке возврата и не возобновляется, пока этот цикл виден в окне статуса. Этот режим занимает минимально возможную часть времени цикла обработки программы.
- В среде тестирования ”**laboratory**” [“лаборатория”] Ваша программа также тестируется в режиме online, но в лабораторных условиях. В этом случае состояние операторов в программных циклах, выполняемых более одного раза в цикле обработки программы процессором, отображается в конце каждого цикла. Этот режим может занимать значительную часть времени цикла обработки программы, зависящую от числа итераций цикла и количества тестируемых команд.

Среду тестирования можно выбрать с помощью команды меню **Debug → Test Environment → Laboratory/Process** [Отладка → Среда тестирования → Лаборатория/Процесс].

Запуск и остановка статуса программы

Статус программы запускается и останавливается с помощью команды меню **Debug → Monitor** [Отладка → Наблюдение]. Статус программы отображается только для области, видимой в данный момент в редакторе.



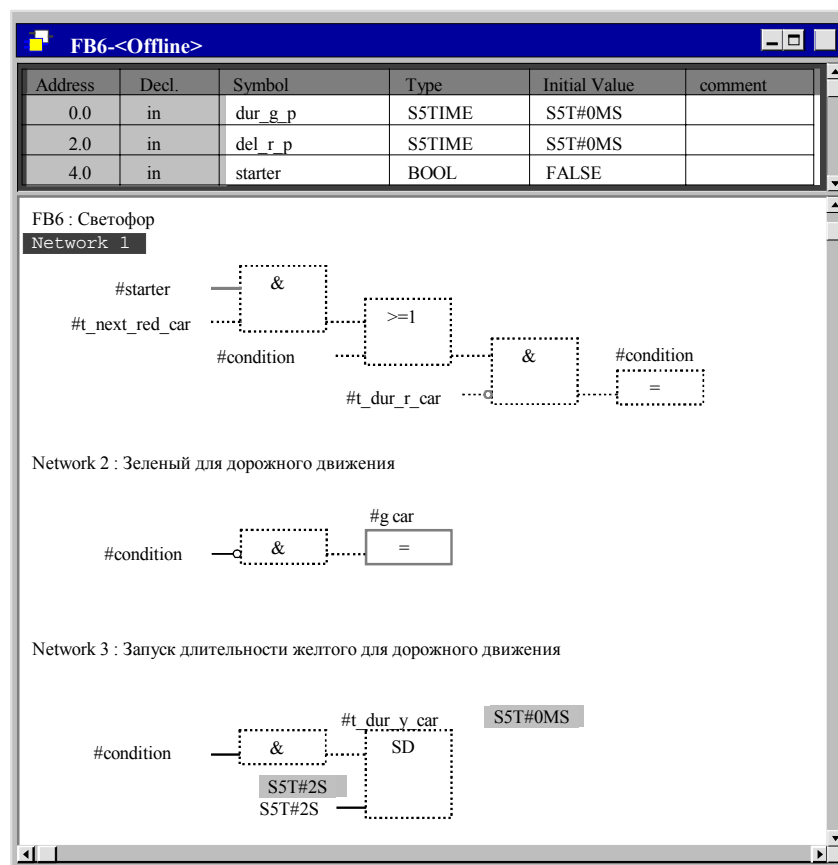


Рис. 5-5. Статус программы в FUP (пример)

Контроль времени цикла

Активизация режима тестирования увеличивает время цикла. Если установленное время цикла превышено, то CPU переходит в состояние STOP, если Вы не запрограммировали OB 80.

Вы можете отобразить и проконтролировать установленное в данный момент время цикла, используя команду меню **PLC → Module Information** [Контроллер → Информация о модуле]. Если необходимо, Вы можете изменить максимальное время цикла в свойствах CPU для целей тестирования при назначении параметров аппаратного обеспечения.

