

Введение

2

В этой главе

Эта глава кратко описывает структуру программы пользователя, состоящую из блоков.

Редактор FUP работает на платформе SIMATIC Manager, который лежит в основе всех приложений. В этой главе объясняется, как попасть из SIMATIC Manager в редактор FUP и как создавать блоки, встроенные в структуру проекта.

Обзор главы

В разделе	Вы найдете	на стр.
2.1	Структура программы пользователя	2–2
2.2	Создание программы пользователя. Обзор	2–4
2.3	Правила, которые необходимо соблюдать	2–7

2.1 Структура программы пользователя

Логические блоки и блоки данных

Программа пользователя состоит из логических блоков и блоков данных. Логические блоки - это блоки, содержащие кодовую часть, например, организационные блоки, функциональные блоки и функции.

Организационные блоки

Организационные блоки (ОВ) представляют собой мост между операционной системой и программой пользователя. Различные организационные блоки выполняют разные функции. Для создания прикладной программы на FUP для своего CPU S7 Вы выбираете организационные блоки, необходимые для Вашей конкретной задачи автоматизации. Для выполнения основной задачи Вам потребуются блоки:

- запуска (OB100, OB101)
- циклической обработки (OB1)
- обработки ошибок (от OB80 до OB87, OB121, OB122), если Вы не хотите, чтобы CPU переключался в STOP при возникновении ошибок.

Имеются также организационные блоки для обработки прерываний в CPU или прерываний от процесса.

За подробной информацией о функциях каждого организационного блока и стартовой информацией, предоставляемой операционной системой CPU, обращайтесь к *Справочному руководству /235/*.

Функции / функциональные блоки

Вы можете программировать каждый организационный блок (ОВ) как структурную программу, создавая функции (FC) и функциональные блоки (FB) и вызывая их в кодовой части ОВ. При вызове блоков Вы снабжаете их данными, требуемыми для объявленных параметров.

- Функциональный блок (FB) - это логический блок "с памятью". В качестве памяти служит при этом соответствующий функциональному блоку экземпляр блока данных, в котором хранятся фактические параметры и статические данные функционального блока.
- Функция (FC) - это логический блок "без памяти", иными словами, без соответствующего экземпляра DB. После обработки FC его выходные параметры содержат рассчитанные значения функции. Дальнейшее использование и сохранение фактических параметров после вызова полностью зависит от пользователя.

Данные

Операционная система делает доступными следующие данные

- Периферийные входы и выходы
- Образ процесса на входах и выходах
- Меркеры
- Таймеры
- Счетчики

Вы можете также определить свои собственные данные:

- Вы можете определить в блоках данных разделяемые (глобальные) данные. Эти данные доступны всей программе пользователя.

- Статические переменные действительны только в функциональном блоке, где они определены. При каждом вызове функционального блока указывается экземпляр блока данных, который кроме всех параметров содержит также статические данные. Если определена многоэкземплярная модель, то экземпляры данных, включая статические данные, хранятся в экземпляре блока данных.
- При создании логического блока Вы можете определить временные данные. Эти данные требуют только стековой памяти во время текущей обработки блока.

Блоки данных

Блоки данных хранят данные программы пользователя. Имеются два типа блоков данных: разделяемые (глобальные) блоки данных и экземпляры блоков данных.

- К разделяемым блокам данных возможен доступ из всех блоков в программе.
- Экземпляры блоков данных ставятся в соответствие функциональным блокам и содержат помимо данных FB также данные определенных при необходимости мультиэкземпляров. Поэтому Вы должны обращаться к экземплярам блоков данных только в связи с этими функциональными блоками.

Дополнительная информация

Введение в методику программирования Вы найдете в *Руководстве по программированию /234/*.

2.2 Создание программы пользователя. Обзор

Программа пользователя

Программа пользователя, работающая на CPU S7, в основном состоит из блоков. Она содержит также такую информацию, как данные о конфигурации системы и о ее включении в сеть. В зависимости от Вашего приложения программа пользователя будет включать в себя следующие элементы:

- организационные блоки (OB)
- функциональные блоки (FB)
- функции (FC)
- блоки данных (DB)

Для облегчения работы Вы можете создавать свои собственные типы данных, определенные пользователем (UDT – User-defined Data Type), которые могут использоваться или как типы данных в собственном смысле, или как шаблон для создания блоков данных.

Некоторые из часто используемых блоков, такие, как системные функциональные блоки (SFB) и системные функции (SFC), встроены в CPU. Другие блоки (например, блоки для функций IEC или блоки управления по замкнутому контуру) имеются в виде отдельных пакетов. Эти блоки не нужно программировать, их просто загружают в программу пользователя.

Указание

Вы можете проверить, какие SFB и SFC встроены в Ваш CPU, щелкнув мышью в режиме online на команде меню **PLC → Module Information** [Контроллер → Информация о модуле]

Инкрементный редактор FUP

Стандартное программное обеспечение STEP 7 включает в себя редактор для программирования блоков. Этот редактор может быть установлен на язык программирования FUP для программирования логических блоков (OB, FB, FC). Редактор FUP инкрементно. Это значит, что каждый Ваш ввод сопровождается синтаксическим контролем. При этом синтаксические ошибки отображаются, неправильное размещение элементов FUP и ошибочные адреса отвергаются немедленно.

Запуск из SIMATIC Manager

Редактор FUP запускается из SIMATIC Manager. Прежде чем Вы сможете это сделать, Вы должны сначала создать в SIMATIC Manager проект, содержащий S7-программу. S7-программу можно создать так, чтобы она зависела или не зависела от аппаратных средств. Вы можете или вставить S7-программу непосредственно в проект, или редактировать S7-программу, соответствующему программируемому модулю. Сама программа может служить в качестве контейнера для блоков, файлов с исходными текстами или планов.

Используя редактор FUP, Вы можете работать только с блоками, хранящимися в папке "Blocks" ["Блоки"].

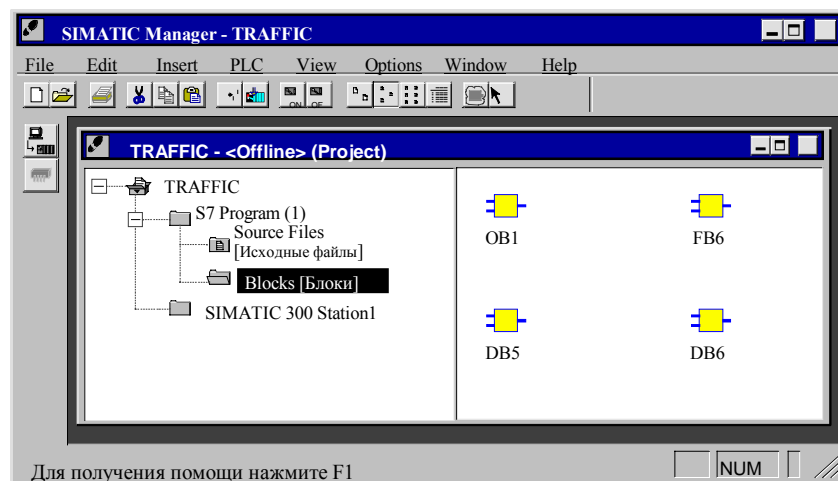


Рис. 2-1. Запуск редактора FUP из SIMATIC Manager'a

Пояснение к рис. 2-1: File -Файл, Edit - Редактировать, Insert - Вставить, PLC - Контроллер, View - Вид, Options - Параметры, Window - Окно, Help - Помощь

Создание блока

Для создания первого блока сначала необходимо создать в SIMATIC Manager'e пустой блок, с помощью которого Вы затем сможете открыть редактор. Открыв редактор FUP, Вы затем сможете создавать другие блоки.

- В SIMATIC Manager'e Вы можете выбрать папку "Blocks" ["Блоки"] и вставить желаемый тип блока, выбрав **Insert** → **S7 Block ...** [Вставить → Блок S7 ...]. Новый блок появляется с правой стороны окна проекта.
- Оказавшись в редакторе, Вы можете создать новый блок, выбрав **File** → **New** [Файл → Новый]. В появившемся диалоговом окне Вам предлагается определить требуемый тип и номер блока.

Выбор языка программирования

При создании блока Вы одновременно выбираете язык программирования, который Вы хотите использовать. На основе этого выбора открывается соответствующий редактор. Для программирования на FUP выберите его в качестве рабочего языка следующим образом:

- В SIMATIC Manager'e в диалоговом окне "Properties" ["Свойства"]
- В редакторе FUP в диалоговом окне "LAD/STL/FBD" [в немецком варианте - "KOP/AWL/FUP"] в закладке "Editor" ["Редактор"], которая выбирается в строке меню через **Options** → **Customize** [Параметры → Настройка].

Открытие блока

В SIMATIC Manager'e блок можно открыть двойным щелчком на этом блоке. Другие способы: выбор команды меню **Edit** → **Open Object** [Редактировать → Открыть объект] или щелчок на соответствующей кнопке на панели инструментов.

Сохранение и загрузка блоков

При сохранении блока в редакторе помните следующее:

- **File** → **Save** [Файл → Сохранить] всегда сохраняет блок в папке "Blocks" ["Блоки"] на жестком диске Вашего устройства программирования или PC.
- **PLC** → **Download** [Контроллер → Загрузить] загружает открытый блок в CPU.

Создав блоки для своей программы, загрузите их в CPU S7, используя SIMATIC Manager. За дополнительной информацией о загрузке программ пользователя обращайтесь к *Руководству пользователя*

Загрузка созданных блоков по отдельности в CPU не всегда достаточна, так как иногда могут потребоваться данные о конфигурации системы. Поэтому следует загружать полную программу в SIMATIC Manager'e.

Вызов вспомогательных функций

Редактор FUP имеет следующие функции, которые могут оказаться полезными при создании и запуске программ.

Таблица 2–1. Вспомогательные функции в редакторе FUP

Функция	Команда меню
Вызов справочных данных о программе пользователя	Options → Reference Data [Параметры → Справочные данные]
Редактирование таблицы символов или отдельных символов	Options → Symbol Table [Параметры → Таблица символов] или Options → Edit Symbols [Параметры → Редактировать символы]
Наблюдение/управление переменными	PLC → Monitor/Modify Variables [Контроллер → Наблюдение/управление переменными]
Отображение/изменение режима работы или сброс памяти CPU	PLC → Operating Mode [Контроллер → Режим работы] или PLC → Clear/Reset [Контроллер → Сбросить]
Отображение состояния выбранного модуля	PLC → Module Information [Контроллер → Информация о модуле]
Установка времени и даты CPU	PLC → Set Time and Date [Контроллер → Установить время и дату]

Эти функции подробно описаны в *Руководстве пользователя* /231/.

2.3 Правила, которые необходимо соблюдать

Порядок создания блоков

Порядок, в котором создаются логические блоки и блоки данных в программе пользователя, имеет важное значение. Как правило: **если блоки вызываются внутри других блоков, то вызываемые блоки уже должны существовать перед тем, как программируются их вызовы.** Ввод несуществующего блока в качестве элемента FUP невозможен. Если несуществующий блок программируется с использованием команды CALL, то сообщение об ошибке появляется при сохранении программы, так как вызываемый блок не может быть найден.

Редактирование при исполнении программы

Используя STEP 7, Вы можете редактировать программу пользователя, хранящуюся в CPU, в режиме online, когда CPU находится в RUN.



хранящейся в CPU, может привести к ситуациям, при которых машины и устройства внезапно включаются или выключаются, при этом возможно травмирование персонала или повреждение оборудования.

Предупреждение

Модификация работающей программы в режиме online может привести к неправильному функционированию и непредвиденным реакциям в Вашей установке, что может вызвать травмы персонала или нанести ущерб оборудованию. Если CPU включен online и находится в режиме RUN, изменение программы пользователя, изменение программы пользователя, при которых машины и устройства внезапно включаются или выключаются, при этом возможно травмирование персонала или повреждение оборудования.

Всегда планируйте последовательность событий в Вашем процессе в соответствии с правилами техники безопасности. Никогда не пытайтесь изменять работающую программу в режиме online, не обдумав сначала последствия своих действий и не приняв надлежащих мер, предотвращающих несчастные случаи.

Указание

За информацией о работе в режимах online и offline обращайтесь к *Руководству пользователя STEP 7 /231/*.
