

# Создание блоков данных и типов данных, определенных пользователем

## 4

### В этой главе

Блоки данных - это важная составная часть Вашей прикладной программы, так как они содержат все ее данные. В этой главе объясняется, как создаются блоки данных.

Типы данных, определенных пользователем (UDT), не существенны для программирования. Однако, они могут сэкономить много времени в ситуациях, когда Вам необходимо писать программы для похожих задач.

### Обзор главы

В разделе	Вы найдете	на стр.
4.1	Создание блоков данных – обзор	4–2
4.2	Выбор метода	4–4
4.3	Редактирование таблицы описаний	4–5
4.4	Редактирование текущих значений данных	4–6
4.5	Создание типов данных, определенных пользователем (UDT)	4–8

## 4.1 Создание блоков данных - Обзор

### Блоки данных

Блоки данных (DB) служат для управления данными. Поэтому в них отсутствует операторная часть. Программирование блоков данных включает в себя следующие части:

- **Таблица описаний:** В таблице описаний Вы устанавливаете структуру данных блока данных.
- **Свойства блока:** Свойства блока содержат такую дополнительную информацию, вносимую системой, как метку времени, язык программирования и путь. Кроме того, Вы сами можете указать имя, семейство, версию и автора и назначить системные параметры для блоков (см. гл. 5).

### Типы блоков данных

Программа пользователя может содержать следующие блоки данных:

- **Глобальные (разделяемые) блоки данных**, к которым можно обращаться из всех логических блоков в программе. Данные продолжают храниться в блоках данных и после их закрытия.

Если Вам нужны несколько глобальных блоков данных с одинаковой структурой, то Вы можете их создать с помощью типов данных, определенных пользователем (UDT). Тогда речь идет о **блоках данных с соответствующим UDT**.

- **Экземпляры блоков данных** связываются с конкретными функциональными блоками и структурируются в соответствии с таблицей описания переменной FB. Вы можете создать экземпляр блока данных только в том случае, если уже существует соответствующий функциональный блок. Здесь речь идет о **блоках данных с соответствующим функциональным блоком**.

### Методы создания блоков данных

В зависимости от вида создаваемого блока данных используются разные методы.

**Глобальные блоки данных** можно создавать следующими способами:

- Определить структуру для отдельного блока данных. Для этого Вы должны определить переменные и типы данных в желаемом порядке. Эта структура применима только к этому DB.
- Определить структуру для блока данных с помощью типа данных, определенного пользователем. В этом случае структура данных UDT определяет и структуру DB. Тип данных, определенный пользователем, может быть поставлен в соответствие многим блокам данных.

Если Вы создаете **экземпляр блока данных**, то

- поставьте в соответствие этому блоку данных существующий функциональный блок. В этом случае раздел описаний функционального блока определяет структуру блока данных. Одному функциональному блоку могут быть поставлены в соответствие несколько экземпляров блоков данных.

---

#### Указание

Если Вы изменяете раздел описаний FB, то Вы должны вновь создать все связанные с ним экземпляры блоков данных, чтобы обеспечить их совместимость. То же самое относится и к блокам данных, которые были созданы на основе UDT.

---

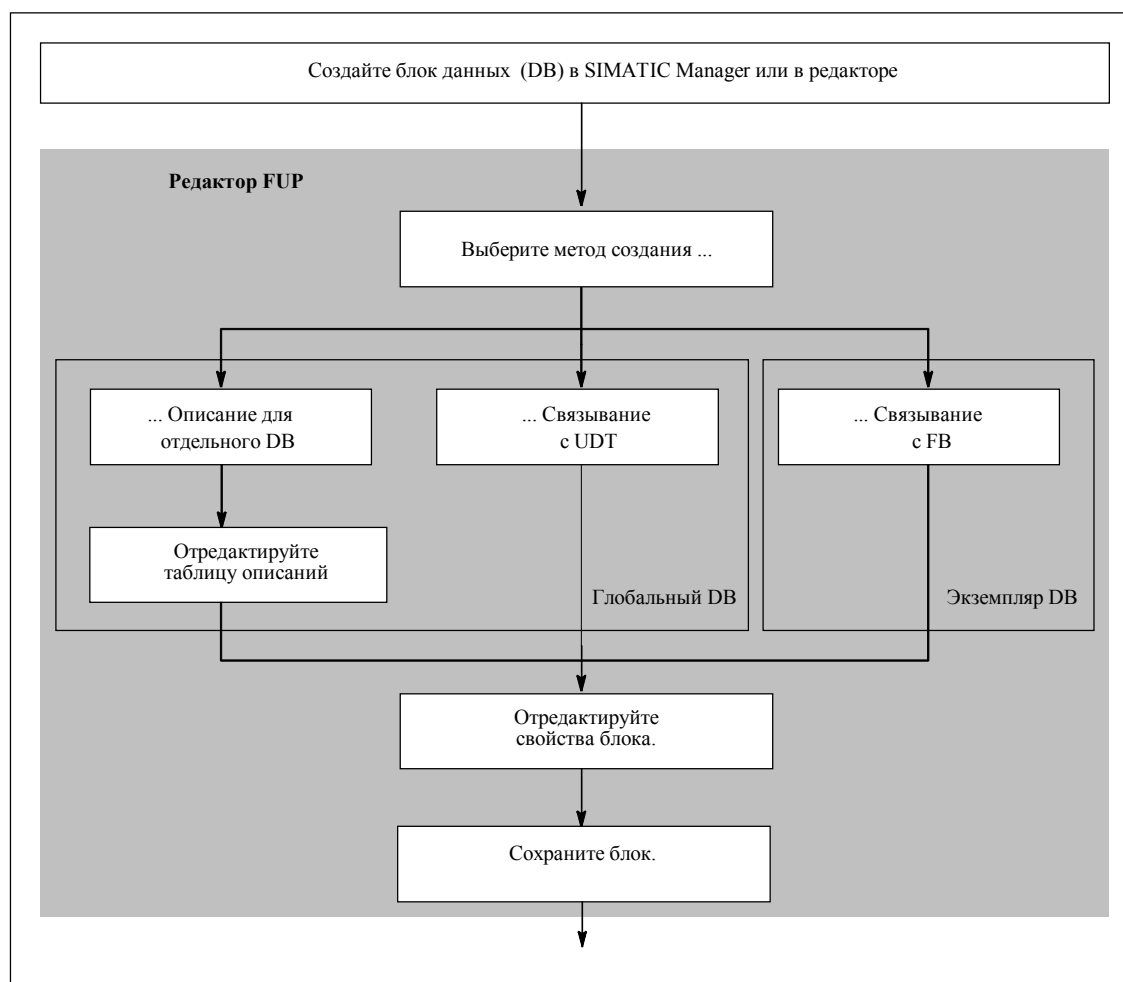


Рис. 4-1. Последовательность действий при программировании блоков данных

## 4.2 Выбор метода

### Последовательность действий

При создании блока данных в SIMATIC Manager или в редакторе FUP Вы должны указать метод, который Вы желаете использовать. Вам будет предложено выбрать метод в диалоговом окне.

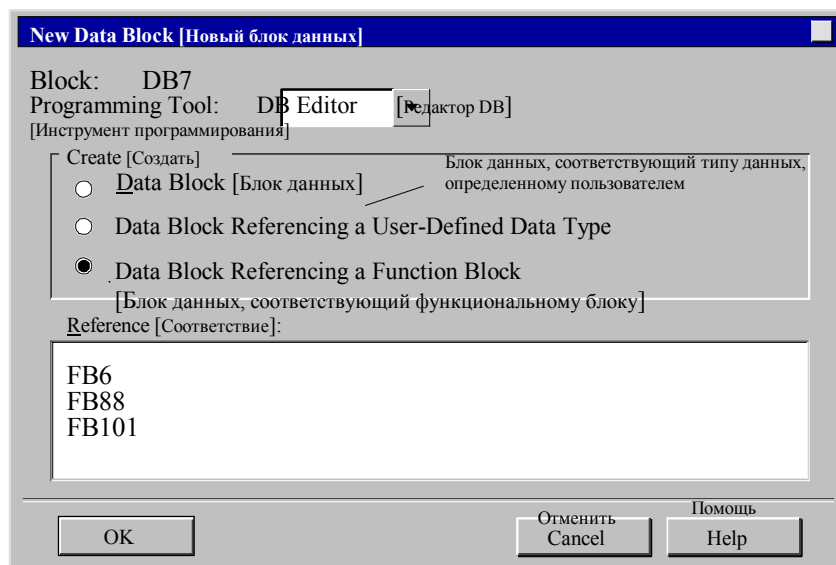


Рис. 4-2. Выбор метода и соответствие FB или UDT

При создании DB, основанного на UDT, или в качестве экземпляра блока данных, соответствующего FB, Вы делаете выбор из окна со списком, отображающего все существующие UDT и FB. UDT или FB уже должен существовать.

### Дальнейшие действия

Дальнейшие действия зависят от того, создаете ли Вы DB путем сопоставления или с помощью отдельного описания.

- Так как соответствующий UDT или FB определяет структуру блока данных, то Вы фактически уже создали новый блок данных. Таблица описаний отображается на экране, но ее нельзя редактировать.
- Если Вы определяете структуру глобального блока данных, то Вы теперь должны отредактировать таблицу описаний, объявляющую имена переменных и типы данных и, если необходимо, начальные значения и комментарии (см. раздел 3.3).

## 4.3 Редактирование таблицы описаний

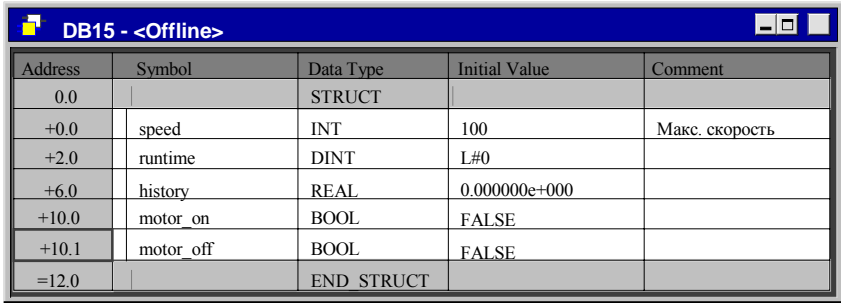
### Назначение окна описаний

При создании отдельного глобального блока данных или UDT Вы должны описать их элементы (переменные) и их типы данных. Для этого используется таблица описаний в окне описаний. При работе с блоками данных Вы можете перейти в это окно командой меню **View → Declaration View** [Вид → Окно описаний].

Это не относится к блокам данных, соответствующим UDT или FB, так как это описание уже определено соответствующим UDT или FB.

### Структура таблицы в окне описаний

Окно описаний блока данных отображает адреса, типы описаний (только для экземпляров DB), имена переменных (символы), начальные значения и комментарии. Пример показан на рис. 4-3:



Address	Symbol	Data Type	Initial Value	Comment
0.0		STRUCT		
+0.0	speed	INT	100	Макс. скорость
+2.0	runtime	DINT	L#0	
+6.0	history	REAL	0.000000e+000	
+10.0	motor_on	BOOL	FALSE	
+10.1	motor_off	BOOL	FALSE	
=12.0		END STRUCT		

Рис. 4-3. Описание блока данных

Столбцы имеют то же значение, что и в таблице описаний логического блока (см. раздел 3.3).

### Последовательность действий

Для ввода нового описания напечатайте требуемый тип описания, имя переменной, тип данных, начальное значение (не обязательно) и комментарий (не обязательно). Вы можете перемещать курсор из одной ячейки в следующую, используя клавишу TAB или RETURN. При достижении конца строки переменной присваивается адрес.

Синтаксис проверяется после редактирования каждой ячейки, ошибки отображаются красным цветом. При этом Вы можете продолжать ввод и исправить ошибки позднее.

#### Указание

Редактирование окна описаний не отличается от редактирования таблицы описания переменных логического блока (см. раздел 3.4). Процедуры редактирования и ввода идентичны, и Вы должны так же действовать и при вводе массивов и структур.

## 4.4 Редактирование текущих значений данных

### Начальное значение /текущее значение

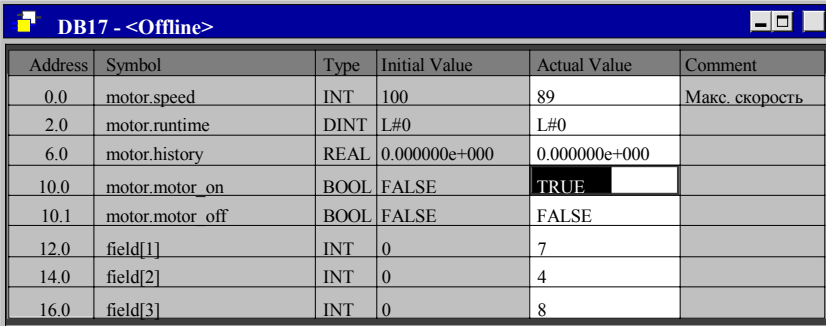
При создании и первом сохранении блока данных объявленное (не обязательное) начальное значение автоматически используется в качестве текущего значения переменной. При обращении к блоку данных программа пользователя продолжает его использовать как текущее значение, пока Вы явно не определите новое текущее значение для переменной в программе пользователя.

Текущие значения переменных изменяются логическими блоками, которые производят запись в них при исполнении программы CPU. Вы имеете возможность отобразить текущие значения переменных и изменить их.

### Окно данных блоков данных

Вам необходимо переключиться в окно данных, чтобы отобразить и редактировать текущие значения данных в блоках данных. Для переключения в окно данных откройте блок данных и используйте команду меню **View → Data View** [Вид → Окно данных].

Единственная разница между окном данных и окном описаний блока данных состоит в дополнительном столбце "Actual Value" ["Текущее значение"]. В окне данных элементы переменных составного типа отображаются отдельно и с их полным символическим именем, так что каждое из их текущих значений может быть отображено и отредактировано (см. рис. 4-4).



Address	Symbol	Type	Initial Value	Actual Value	Comment
0.0	motor.speed	INT	100	89	Макс. скорость
2.0	motor.runtime	DINT	L#0	L#0	
6.0	motor.history	REAL	0.000000e+000	0.000000e+000	
10.0	motor.motor_on	BOOL	FALSE	TRUE	
10.1	motor.motor_off	BOOL	FALSE	FALSE	
12.0	field[1]	INT	0	7	
14.0	field[2]	INT	0	4	
16.0	field[3]	INT	0	8	

Рис. 4-4. Блок данных в окне данных

### Показанное текущее значение

Показанное текущее значение - это значение, которое переменная имела к моменту открытия блока данных, или самое последнее измененное и сохраненное значение.

#### Замечание

В блоках данных, открытых в режиме online, текущее значение циклически не обновляется.

### Изменение и повторная инициализация текущих значений

Вы можете переписать текущие значения в столбце "Actual Value" ("Текущее значение"). Введенные значения должны быть совместимы с типом данных.

С помощью команды меню **Edit → Initialize Data Block** [Редактировать → Инициализировать блок данных] Вы можете повторно инициализировать весь блок данных. При текущие значения переменных заменяются начальными значениями, которые Вы объявили в окне описаний, или теми, которые были объявлены в соответствующем FB или UDT.



## Сохранение текущих значений

Текущие значения становятся действительными, когда Вы их сохраняете.

- Для сохранения текущих значений данных, которые Вы изменили в режиме offline, выберите команду меню **File → Save** [Файл → Сохранить] или щелкните на кнопке "Save" [Сохранить] на панели инструментов. Даже если блок данных был открыт online, будет сохранен блок данных, существующий offline.
- Для загрузки измененных значений данных в CPU выберите команду меню **PLC → Download** [Контроллер → Загрузить] или щелкните на соответствующей кнопке на панели инструментов.



## 4.5 Создание типов данных, определенных пользователем (UDT)

### Обзор

Типы данных, определенные пользователем, - это созданные Вами структуры данных, которые сохраняются в виде блоков. Однажды определив их, Вы можете их использовать под их абсолютными или символическими именами во всей программе пользователя. Вы можете использовать UDT следующим образом:

- как элементарные или составные **типы данных** в разделе описаний логических блоков (FC, FB, OB) или блоков данных (DB)
- как **шаблоны для создания блоков данных** с такой же структурой данных.

### Последовательность действий

На рис. 4–5 показана принципиальная последовательность действий для создания типа данных, определенного пользователем:

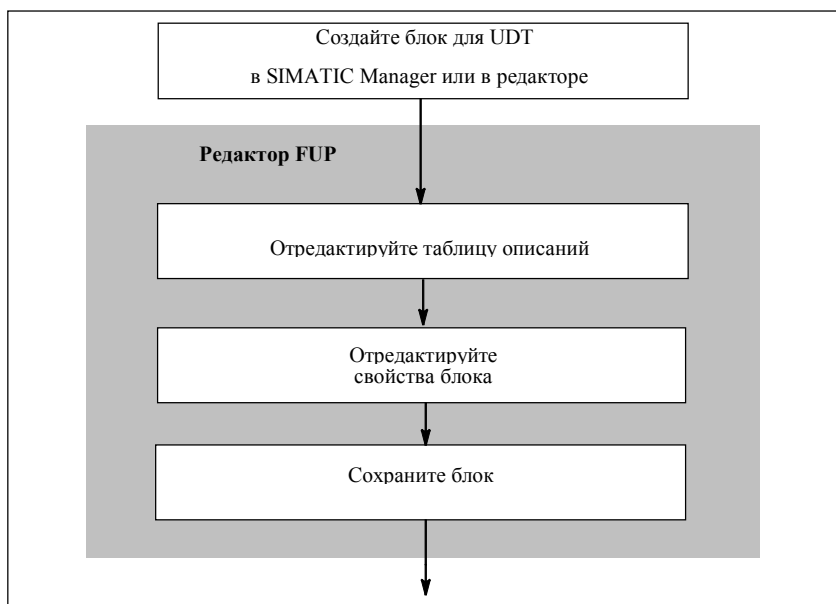
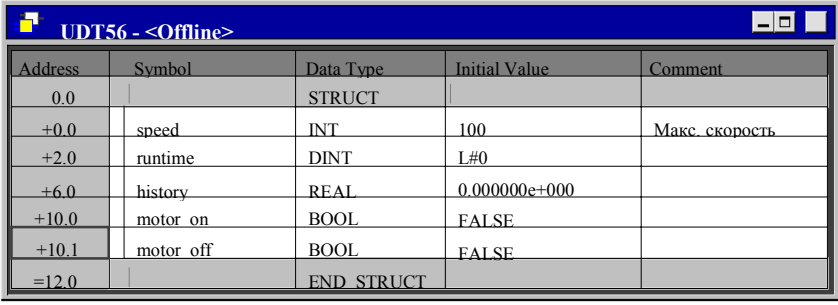


Рис. 4-5. Создание типа данных, определенного пользователем

### Редактирование таблицы описаний

После создания или открытия UDT в SIMATIC Manager или в инкрементном редакторе на экране отображается таблица описаний, в которой Вы должны задать структуру типа данных.





Address	Symbol	Data Type	Initial Value	Comment
0.0		STRUCT		
+0.0	speed	INT	100	Макс. скорость
+2.0	runtime	DINT	L#0	
+6.0	history	REAL	0.000000e+000	
+10.0	motor_on	BOOL	FALSE	
+10.1	motor_off	BOOL	FALSE	
=12.0		END_STRUCT		

Рис. 4-6. Описание UDT

Первая и последняя строка окна описания для UDT заранее заданы и содержат ключевые слова STRUCT и END\_STRUCT для начала и конца типа данных, определенного пользователем. Эти строки не могут редактироваться.

Вначале на экране отображаются две пустые строки, дающие Вам возможность описывать свои переменные. Вы должны ввести имя переменной и тип данных. Начальные значения и комментарии не обязательны. Вы можете создать новые пустые строки, используя команду меню **Insert → Declaration Row → Before Selection / After Selection** [Вставить → Строка описания → Перед выделением/После выделения].

---

#### Указание

Редактирование этой таблицы описаний аналогично редактированию таблицы описаний логических блоков и блоков данных.

---