

Создание блоков данных и

типов данных, определенных пользователем

4

**Что описывает
эта глава?**
как их можно создать.

Блоки данных - это важная составная часть Вашей прикладной программы, так как в них хранятся данные прикладной программы. В этой главе описано,

Типы данных, определенных пользователем, напротив, скорее являются удобством и не нужны в обязательном порядке для программирования. Однако, если Вы часто создаете программы для похожих задач, то Вы вскоре начнете их использовать, чтобы сэкономить массу времени с помощью “собственных” типов данных.

Обзор главы

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

4.1. Создание блоков данных - Обзор

Блоки данных

Блоки данных (DB) служат для управления данными. Поэтому в них отсутствует операторная часть. Поэтому при программировании Вы должны принимать во внимание соответственно 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 разработки или указание пути. Кроме того, Вы сами можете указать имя, семейство, версию и автора (см. гл. 5).

Виды DB

В прикладной программе в принципе могут быть следующие типы блоков данных:

- **Глобальные блоки данных**, к которым можно обращаться из всех блоков и которые сохраняют данные длительное время.
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.** Экземпляры DB по мере надобности подчиняются функциональным блокам и структурируются в соответствии с переменными, описанными в FB. Поэтому предпосылкой для создания экземпляра DB является существование FB. Поэтому здесь речь идет о **блоках данных с соответствующим функциональным блоком**.

Методы создания блоков данных

В зависимости от вида создаваемого DB Вы можете использовать различные методы.

Глобальные блоки данных можно создавать следующими способами:

- Определить структуру для отдельного DB: для этого Вы должны определить для DB переменные и типы данных в желаемой последовательности. Эта структура, таким образом, пригодна только для этого блока данных..
- Определить структуру для DB с помощью типа данных, определенного пользователем. В этом случае структура данных типа, определенного пользователем, определяет и структуру DB. Тип данных, определенный пользователем, может быть поставлен в соответствие многим DB.

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

- поставьте в соответствие этому DB существующий FB. В этом случае раздел описаний FB определяет структуру DB. Одному FB могут быть поставлены в соответствие несколько экземпляров DB.

Указание

Если Вы в последствии изменяете описание переменных FB, то Вы должны вновь создать соответствующий экземпляр DB, так как оба блока теперь не подходят друг другу. По смыслу это справедливо и для DB, которые были созданы в соответствии с UDT.

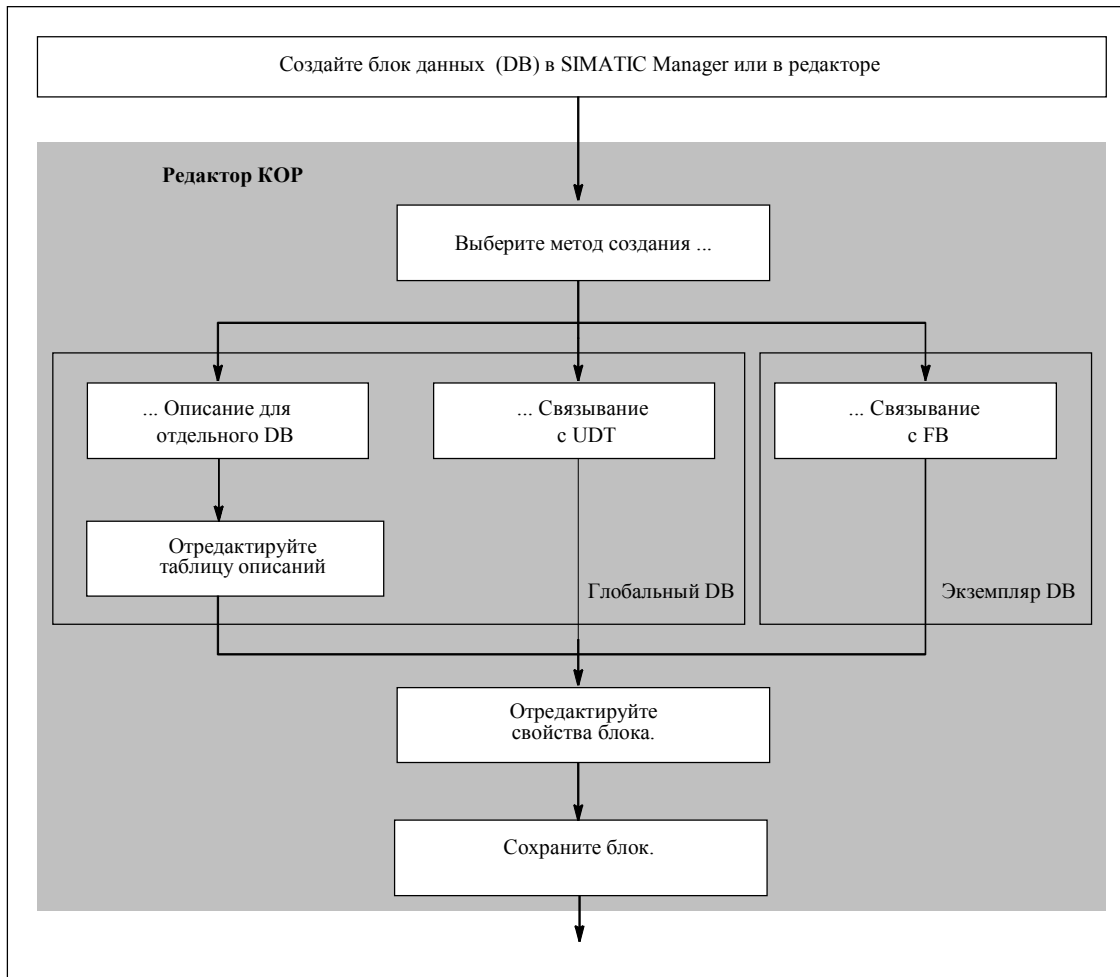


Рис. 4-1. Последовательность действий при программировании блоков данных

4.2. Выбор метода создания

Последовательность действий При создании блока данных в SIMATIC Manager или в редакторе AWL Вы должны указать метод создания. К этому Вас пригласит соответствующий диалог.

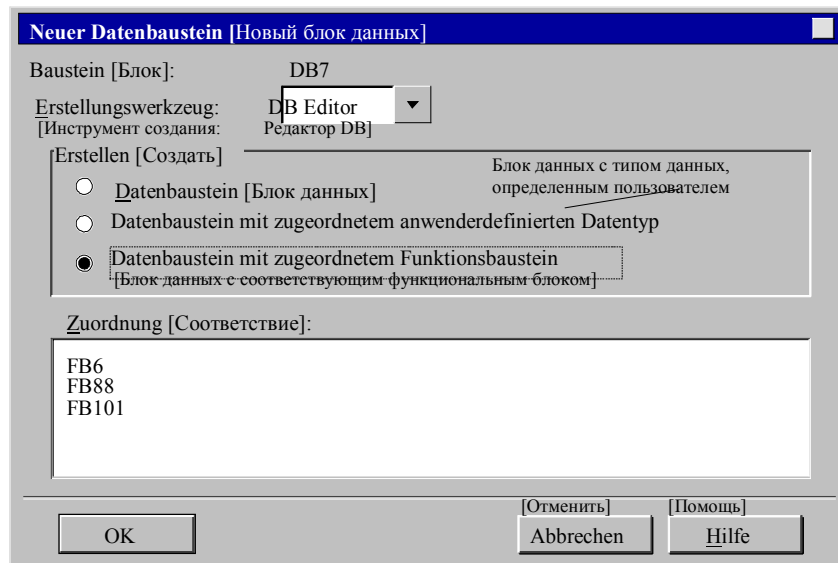


Рис. 4-2. Выбор метода и соответствие FB или UDT

Если Вы хотите создать блок данных с помощью UDT или как экземпляр DB функционального блока, Вы должны сделать выбор в списке, который приводит все UDT или FB программы. Существование соответствующего UDT/FB является, таким образом, безусловной предпосылкой.

Дальнейшие действия

Дальнейшие действия различаются в зависимости от того, создаете ли Вы блок данных путем сопоставления или с помощью отдельного описания.

- Так как соответствующий UDT или FB уже однозначно определяет структуру блока данных, то в этих случаях создание блока данных на этом заканчивается. Для дополнительной проверки отображается таблица описаний, которую, однако, нельзя редактировать.
- Напротив, при отдельном создании глобального блока данных Вы должны теперь продолжить редактирование таблицы описаний, в которой указываются имена переменных и типы данных, а также (не обязательно) начальные значения и комментарии (см. гл. 4.3).

4.3. Редактирование таблицы описаний

Назначение окна описаний

типы

которую Вы в данном случае можете

Deklarationssicht (Вид ▶

При отдельном создании глобального блока данных или при создании UDT Вы должны задать элементы (переменные) DB или UDT и соответствующие данных. Для этого используется таблица, отображаемая в окне описаний, в перейти с помощью команды меню **Ansicht** ▶

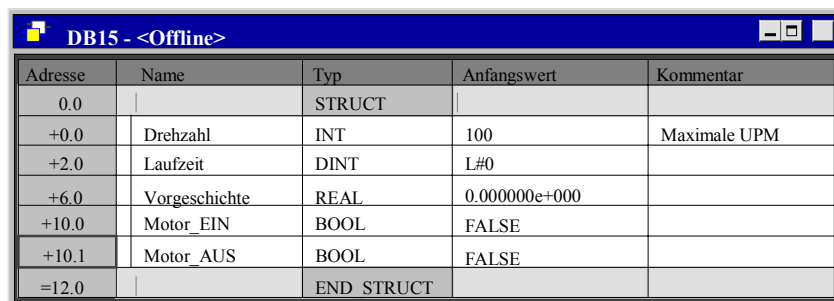
Окно описаний).

У блоков данных, которые были созданы в соответствии с UDT или FB, эта работа отпадает, так как принимается описание соответствующего UDT или FB.

Структура таблицы описаний

показан на рис. 4–3:

Окно описаний блока данных содержит адрес (Adresse), тип описания (только у экземпляров DB), имя переменной (Name) (символ), тип данных (Typ), начальное значение (Anfangswert) и комментарий (Kommentar). **в окне** Пример



Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	Drehzahl	INT	100	Maximale UPM
+2.0	Laufzeit	DINT	L#0	
+6.0	Vorgeschichte	REAL	0.000000e+000	
+10.0	Motor_EIN	BOOL	FALSE	
+10.1	Motor_AUS	BOOL	FALSE	
=12.0		END STRUCT		

Рис. 4-3. Описание блока данных

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

Последовательность действий

При новом вводе описания Вы указываете вслед за желаемым типом описания друг за другом имя переменной, тип данных, начальное значение (не обязательно) и комментарий (не обязательно). При этом курсор передвигается в соседнее поле клавишей TAB или RETURN. По завершении строки переменной присваивается адрес.

После обработки каждого поля таблицы производится **проверка синтаксиса**, имеющиеся ошибки индицируются красным цветом. Однако Вы не обязаны устранять эти ошибки немедленно, а можете продолжить редактирование и выполнить коррекцию позднее.

Указание

Редактирование окна описаний соответствует редактированию таблицы описания переменных кодового блока (см. гл. 3.4). Функции редактирования одинаковы, и при вводе массивов и структур Вы действуете соответствующим образом.

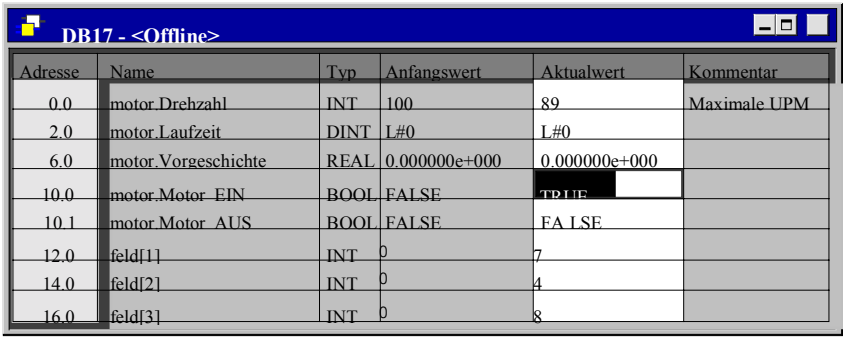
4.4. Редактирование текущих значений данных

Начальное значение - значение Начальное значение, устанавливаемое (не обязательно) при создании блока данных, при первом сохранении этого блока принимается в качестве **текущее** текущего значения для переменной. Соответственно, прикладная программа при первом обращении к блоку работает с этим значением, если только Вы не установили явно другое текущее значение.

Текущие значения переменных при обработке прикладной программы изменяются через кодовые блоки, которые обращаются к переменным для записи. Вы имеете возможность отобразить текущие значения переменных и изменить их.

Окно данных данных Для отображения и редактирования текущих значений данных в блоке **блока** данных Вам нужно окно данных. Для этого откройте блок данных и перейдите в окно данных с помощью команды меню **Ansicht ► Datensicht** (Вид ► Окно данных).

Окно данных отличается от окна описаний блока данных только дополнительным столбцом "Aktualwert" ("Текущее значение"). В окне данных у переменных составного типа элементы перечисляются по отдельности, с полными символическими именами, для того чтобы можно было отобразить и отредактировать текущее значение для каждого элемента (см. рис. 4-4).



Adresse	Name	Typ	Anfangswert	Aktualwert	Kommentar
0.0	motor.Drehzahl	INT	100	89	Maximale UPM
2.0	motor.Laufzeit	DINT	L#0	L#0	
6.0	motor.Vorgeschichte	REAL	0.000000e+000	0.000000e+000	
10.0	motor.Motor_EIN	BOOL	FALSE	TRUE	
10.1	motor.Motor_AUS	BOOL	FALSE	FALSE	
12.0	feld[1]	INT	0	7	
14.0	feld[2]	INT	0	4	
16.0	feld[3]	INT	0	8	

Рис. 4-4. Блок данных / окне данных

Показанное значение Здесь речь идет о значении, которое переменная имела к моменту **текущее** открытия DB, или о значении, которое Вы в последний раз изменили и сохранили.

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

**Изменение и
повторная
инициализация
текущих значений**

В столбце "Aktualwert" ("Текущее значение") Вы можете изменить текущее значение, переписав его. Введенные значения данных должны быть совместимы с типом данных.

С помощью команды меню **Bearbeiten ► Neu initialisieren** (Редактировать ► Повторно инициализировать) Вы можете заново инициализировать весь блок данных. При этом текущие значения переменных переписываются начальными значениями, которые Вы определили в окне описаний или которые определены через соответствующий FB или UDT.

**Сохранение
текущих значений**

Чтобы измененные текущие значения сделать действующим, их нужно сохранить:

- Для сохранения измененных Вами текущих значений данных в режиме offline выберите команду меню **Datei ► Speichern** (Файл ► Сохранить) или щелкните на пиктограмме "Сохранить" ("Speichern"). Даже если Вы открыли блок в режиме offline, заново сохранен будет только не связанный с процессором блок.
- Для загрузки измененных значений данных в CPU выберите команду меню **Zielsystem ► Laden** (Контроллер ► Загрузить) или щелкните на соответствующей пиктограмме на панели инструментов.

4.5. Создание типов данных, определенных пользователем (UDT)

Применение

Типы данных, определенные пользователем, - это созданные Вами структуры данных, которые сохраняются в виде блоков. Поэтому их можно применять во всей прикладной программе, используя их абсолютные или символические имена

- как элементарные или составные **типы данных** в разделе описаний кодового блока (FC, FB, OB) или блока данных (DB) или
- как **шаблон для создания блоков данных** с одинаковой структурой данных.

Последовательность действий Принципиальная последовательность действий показана на рис. 4–5:

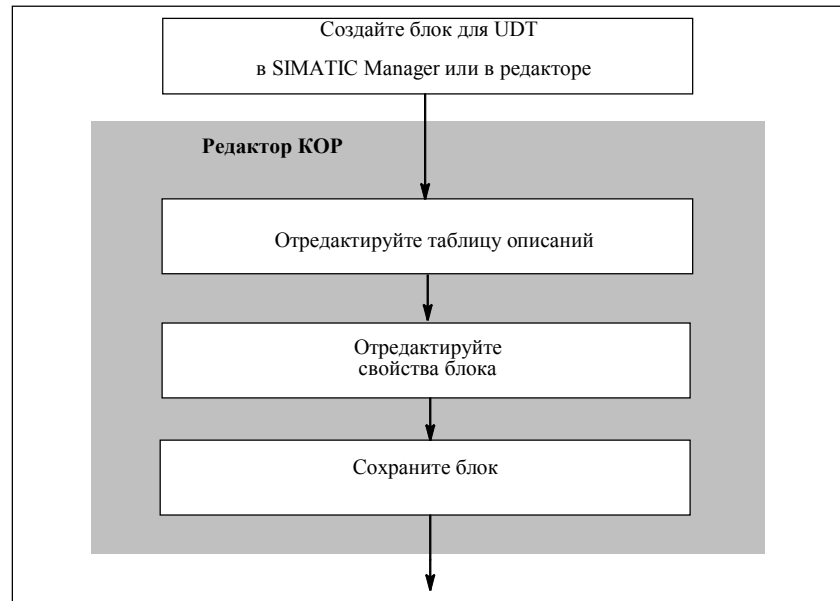


Рис. 4-5. Создание UDT

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.

UDT56 - <Offline>				
Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	Drehzahl	INT	100	Maximale UPM
+2.0	Laufzeit	DINT	L#0	
+6.0	Vorgeschichte	REAL	0.000000e+000	
+10.0	Motor_EIN	BOOL	FALSE	
+10.1	Motor_AUS	BOOL	FALSE	
=12.0		zND 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.

Для ввода переменных Вам сначала предлагаются две пустые строки, в которые Вы вводите по меньшей мере имя и тип данных для двух элементов. Указание начального значения и комментария не обязательно. Новые пустые строки создаются командой меню **Einfügen ▶ Deklarationszeile ▶ Vor Markierung / Nach Markierung** (Вставка ▶ Строка описания ▶ Перед маркировкой/После маркировки).

Указание

Редактирование здесь аналогично редактированию структур в разделе описаний кодовых блоков или блоков данных.
