

# Использование протоколов данных

# 14

STEP 7-Micro/Win предоставляет в ваше распоряжение Мастер протоколирования данных для сохранения результатов измерений процесса в модуле памяти. Перемещение данных процесса в модуль памяти освобождает адреса памяти переменных, которые в противном случае потребовались бы для хранения этих данных.

## В этой главе

Обзор	364
Использование Мастера протоколирования данных	365
Команды, создаваемые Мастером протоколирования данных	369

## Обзор

Поддержка протоколов данных встроена в STEP 7-Micro/WIN и в ПЛК S7-200. С помощью этой функции вы можете постоянно сохранять записи, содержащие данные процесса, под управлением программы. Эти записи могут содержать также метку времени и дату. Вы можете сконфигурировать до четырех независимых протоколов данных. Формат записи протокола данных определяется в новом Мастере протоколирования данных.

Все протоколы данных хранятся в модуле памяти. Для использования функции протоколирования данных вы должны установить в своем ПЛК необязательный модуль памяти емкостью 64 или 256 Кбайт. Информацию о модулях памяти вы найдете в Приложении А.

Для загрузки содержимого протоколов данных в свой компьютер вы должны использовать проводник S7-200.

Пример применения протокола данных показан на рис. 14-1.

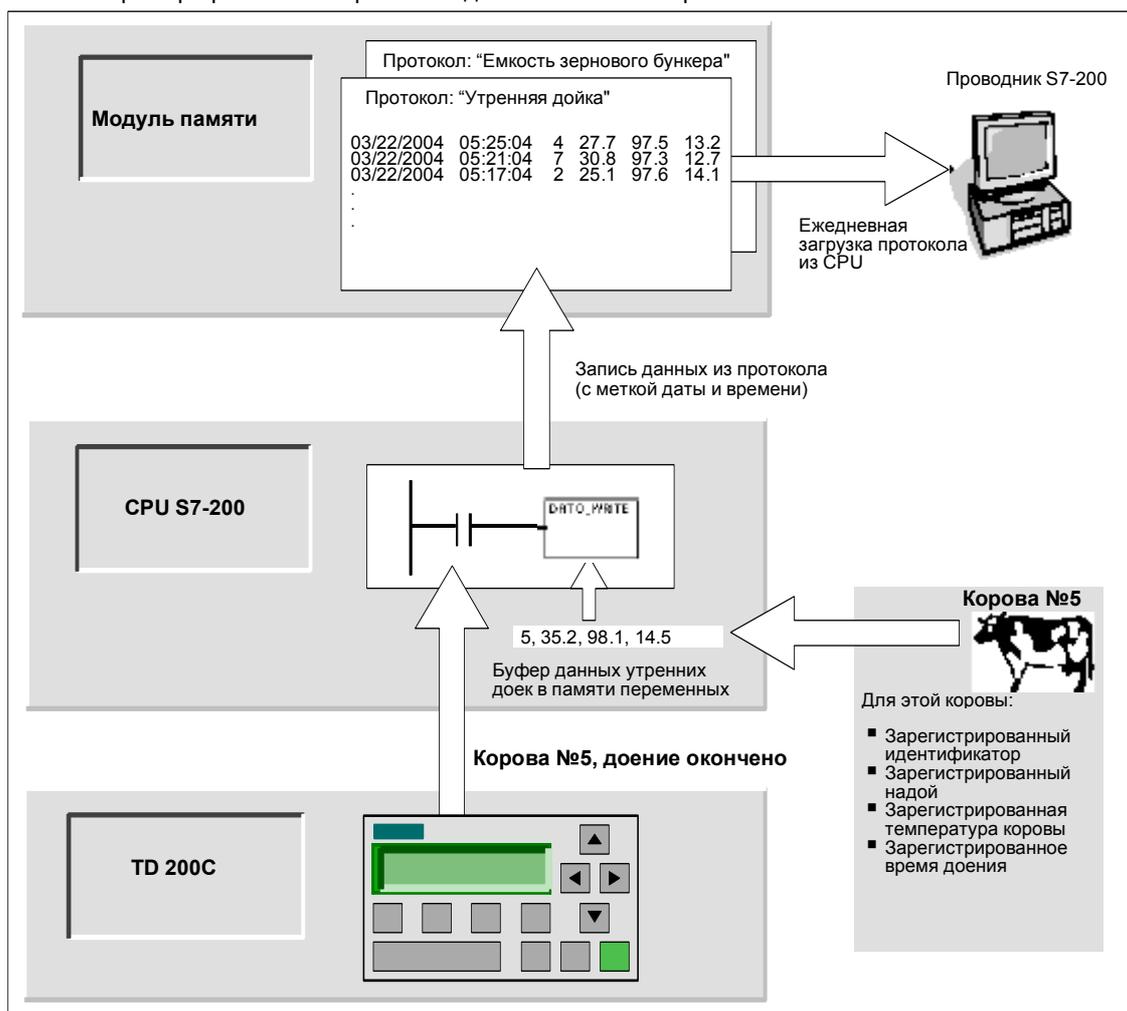


Рис. 14-1. Пример применения протоколирования данных

## Определение протокола данных и терминология

Для лучшего понимания работы Мастера протоколирования данных объясняются следующие определения и понятия.

- Протокол данных – это набор записей данных, обычно упорядоченных по дате и времени. Каждая запись представляет некоторое событие в процессе, которое регистрирует набор данных процесса. Организация этих данных определяется в Мастере протоколирования данных.
- Запись данных протокола – это отдельная строка данных, записанных в протокол.

## Использование Мастера протоколирования данных



Протокол  
данных

В Мастере протоколирования данных можно сконфигурировать до четырех протоколов данных. Мастер протоколирования данных можно использовать для:

- определения формата записи протокола данных
- выбора вариантов протоколирования, например метка времени, метка даты и удаление протокола после загрузки из CPU
- задания максимального количества записей, которые могут храниться в протоколе
- создания кода проекта, используемого для сохранения записей в протоколе данных.

Мастер протоколирования данных создает конфигурацию протокола данных, которая состоит из следующих элементов:

- Таблица символов для каждой конфигурации протоколов данных. Каждая таблица содержит символические имена, которые соответствуют именам полей в протоколе данных. Каждый символ определяет адрес в памяти переменных, по которому сохраняется текущий протокал данных. Каждая таблица содержит также символическую константу для ссылки на отдельные протоколы данных.
- Регистр блока данных для каждой записи протокола данных, в котором каждому полю протокола данных ставится в соответствие адреса в памяти переменных. Ваша программа использует эти адреса памяти переменных для накопления текущего набора данных протокола.
- Подпрограмма-команда DATx\_WRITE. Эта команда копирует указанную запись протокола данных из памяти переменных в модуль памяти. Каждое исполнение команды DATx\_WRITE добавляет новую запись в протокол данных, хранящийся в модуле памяти.

### Опции протокола данных

Вы можете сконфигурировать следующие варианты для протокола данных. См. рис. 14–2.

#### Метка времени

Вы можете снабдить каждую запись протокола данных меткой времени. Если эта опция активизирована, то CPU автоматически снабжает каждую запись данных меткой времени, когда программа пользователя дает команду на запись протокола данных.

#### Метка даты

Вы можете снабдить каждую запись протокола данных меткой даты. Если эта опция активизирована, то CPU автоматически снабжает каждую запись данных меткой даты, когда программа пользователя дает команду на запись протокола данных.

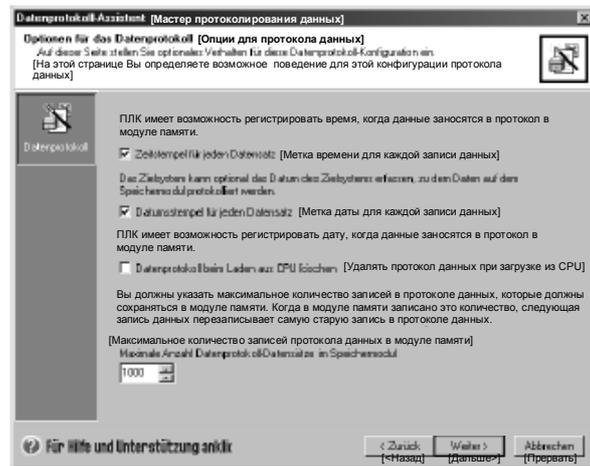


Рис. 14–2. Опции протокола данных

### Очистка протокола данных

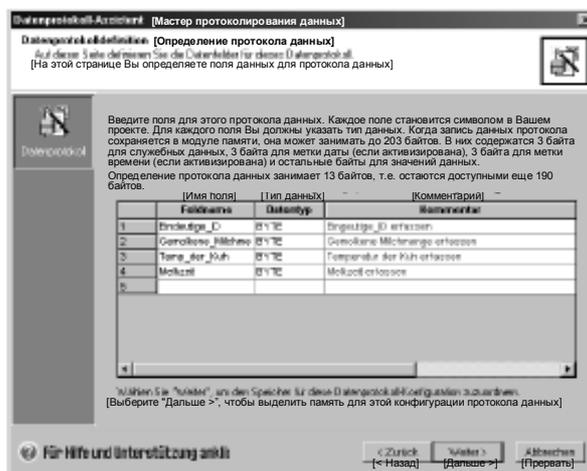
Очистка протокола данных – Вы можете стирать все записи из протокола данных каждый раз, когда он загружается из CPU. Если эта опция установлена, то протокол очищается при каждой его загрузке из CPU.

Протоколы данных реализуются в виде кольцевой очереди (когда протокол полон, новая запись заменяет самую старую). Вы должны указать максимальное количество записей для хранения в протоколе данных. Максимально допустимое количество записей в протоколе данных равно 65 535. Значением по умолчанию для количества записей является 1000.

## Определение протокола данных

Вы задаете поля протокола данных, и каждое поле становится символом в вашем проекте. Для каждого поля вы должны указать тип данных. запись протокола данных может содержать от 4 до 203 байтов данных. Для определения полей данных в протоколе данных действуйте следующим образом. См. рис. 14–3.

1. Для ввода имени щелкните на ячейке Field Name [Имя поля]. Это имя становится символом, на который ссылается программа пользователя.
2. Щелкните на ячейке Data Type [Тип данных] и выберите тип данных из разворачивающегося окна списка.
3. Для ввода комментария щелкните на ячейке Comment [Комментарий].
4. Для определения записи используйте столько строк, сколько необходимо.
5. Щелкните на ОК .



## Редактирование существующих конфигураций протоколов данных

Для редактирования существующих конфигураций протоколов данных действуйте следующим образом:

1. Откройте ниспадающий список конфигураций и выберите существующую конфигурацию протоколов данных, как показано на рис. 14–4.
2. Для удаления существующей конфигурации протоколов данных щелкните на кнопке Delete Configuration [Удалить конфигурацию].

Вы можете иметь до четырех различных протоколов данных.

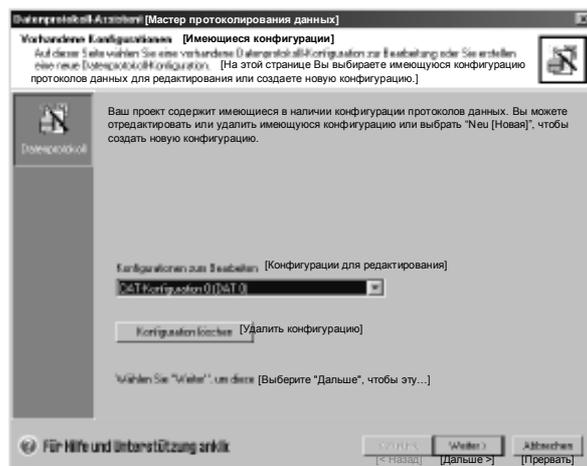


Рис. 14–4. Редактирование существующих конфигураций протоколов данных

## Выделение памяти

Мастер протоколирования данных создает блок в области памяти переменных ПЛК. Этот блок является адресом в памяти, где будет создаваться запись протокола данных перед занесением ее в модуль памяти. Вы указываете начальный адрес в памяти переменных, где вы хотите поместить конфигурацию. Вы можете ввести этот адрес в памяти переменных сами или предоставить возможность Мастеру протоколирования данных предложить адрес неиспользуемого блока памяти переменных нужного размера. Размер блока зависит от выбранных вами опций в Мастере протоколирования данных. См. рис. 14–5.

Для выделения памяти действуйте следующим образом:

1. Для выбора адреса в памяти переменных, где должна быть создана запись протокола данных, щелкните в области Suggested Address [Предлагаемый адрес] и введите этот адрес.
2. Чтобы позволить Мастеру протоколирования данных выбрать неиспользуемый блок в памяти переменных нужного размера, щелкните на кнопке Suggest Address [Предложить адрес].
3. Щелкните на кнопке Next [Дальше].



Рис. 14–5. Выделение памяти

## Компоненты проекта

В диалоговом окне компонентов проекта приведены различные компоненты, которые будут добавлены к вашему проекту. См. рис. 14–6.

Щелкните на Finish [Закончить], чтобы завершить работу с Мастером протоколирования данных и добавить эти компоненты.

Каждой конфигурации протоколов данных может быть присвоено уникальное имя. Это имя отображается в дереве проекта с отдельными конфигурациями Мастера. Определение протокола данных (DATx) присоединяется к концу этого имени.

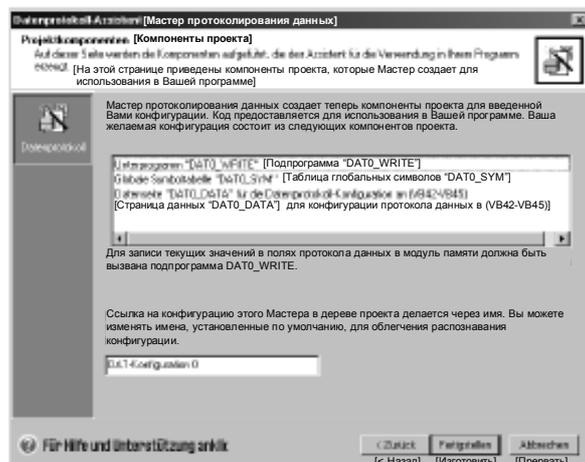


Рис. 14–6. Компоненты проекта

## Использование таблицы символов

Таблица символов создается для каждой конфигурации протоколов данных. Каждая таблица определяет постоянные значения для отдельных протоколов данных. Эти символы могут использоваться в качестве параметров для команд DATx\_WRITE. Кроме того, в каждой таблице создаются символические имена для каждого поля протокола данных. Вы можете использовать эти символы для доступа к значениям протокола данных в памяти переменных.

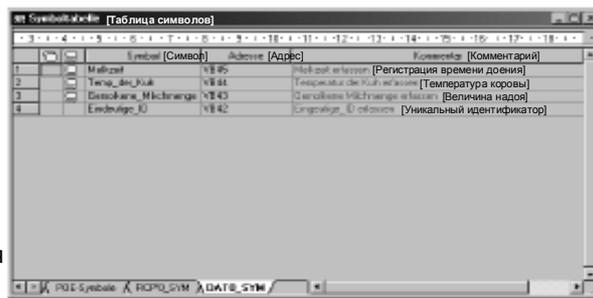


Рис. 14–7. Таблица символов

## Загрузка проекта, содержащего конфигурацию протоколов данных

Перед тем, как использовать протокол данных, Вы должны загрузить проект, содержащий конфигурацию протоколов данных, в CPU S7–200. Если проект содержит конфигурацию протоколов данных, то в диалоговом окне для загрузки в CPU опция для конфигураций протоколов данных по умолчанию активизирована.



### Совет

При загрузке в ПЛК проекта с конфигурациями протоколов данных все записи протоколов данных, хранящиеся в данный момент в модуле памяти, теряются.

Для загрузки проекта, содержащего конфигурацию протоколов данных, действуйте следующим образом. См. рис. 14–8.

1. Выберите команду меню **File > Download [Файл > Загрузить]**.
2. В диалоговом окне в разделе Options [Опции] активизируйте триггерную кнопку Data Log Configuration [Конфигурация протоколов данных].
3. Щелкните на кнопке Download [Загрузить].

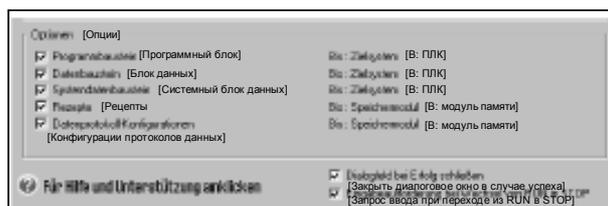


Рис. 14–8. Загрузка проекта с конфигурацией протоколов данных

## Использование проводника S7–200

Проводник S7–200 – это приложение, используемое для чтения протокола данных из модуля памяти, а затем сохранения этого протокола в CSV-файле (в файле, в котором значения разделены запятыми).

При каждом чтении протокола данных создается новый файл. Этот файл сохраняется в каталоге Data Log [Протокол данных]. имя файла имеет следующий формат: адрес ПЛК, имя протокола данных, дата и время.

Вы можете выбрать, должно ли приложение, связанное с расширением CSV, автоматически запускаться при успешном считывании протокола данных. Этот выбор можно сделать через контекстное меню правой клавиши мыши для файла протокола данных.

Каталог протоколов данных находится в каталоге, указанном при инсталляции. Если STEP7 не установлен, то по умолчанию каталогом инсталляции является c:\program files\siemens\Microsystems. Если STEP7 установлен, то каталогом инсталляции по умолчанию является c:\siemens\Microsystem.

Для чтения протокола данных действуйте следующим образом:

1. Откройте проводник Windows. Должна автоматически появиться папка My S7–200 Network [Моя сеть S7–200].
2. Выберите папку My S7–200 Network.
3. Выберите папку соответствующего ПЛК S7–200.
4. Выберите папку модуля памяти
5. Найдите нужный файл конфигурации протоколов данных. Эти файлы называются DAT Configuration x (DATx).
6. Вызовите правой кнопкой мыши контекстное меню, а затем выберите Upload [Загрузить из CPU].

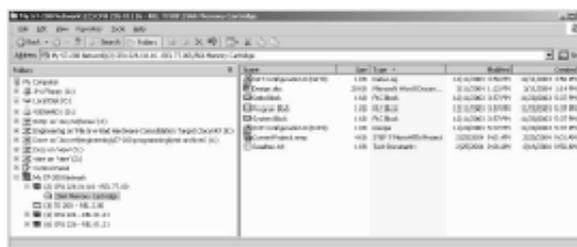


Рис. 14–9. Использование проводника S7–200

## Команды, создаваемые Мастером протоколирования данных

Мастер протоколирования данных добавляет в ваш проект одну подпрограмму-команду.

### Подпрограмма DATx\_WRITE

Подпрограмма DATx\_WRITE записывает текущие значения полей протокола данных в модуль памяти. DATx\_WRITE добавляет одну запись к запротоколированным данным в модуле памяти. Вызов этой подпрограммы осуществляется следующим образом.

Если этой команде не удастся правильно обратиться к модулю памяти, то возвращается ошибка 132.

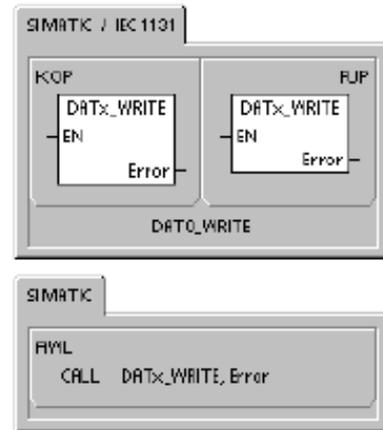


Таблица 14–1. Параметры подпрограммы DATx\_WRITE

Входы/выходы	Тип данных	Операнды
Error	Byte	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD



#### Совет

ЭСППЗУ, используемое в модуле памяти, поддерживает ограниченное количество операций записи. Обычно оно составляет один миллион циклов записи. Когда этот предел достигнут, ЭСПЗУ перестает работать надлежащим образом.

Обратите внимание на то, чтобы команда DATx\_WRITE не выполнялась в каждом цикле обработки программы. Активизация этой команды в каждом цикле приводит к износу модуля памяти за относительно короткий интервал времени.

