

# SIMATIC

## Системные и стандартные функции для S7-300 и S7-400

### Справочное руководство

Данное руководство является частью пакета документации с заказным номером:

6ES7810-4CA07-8BW1

Выпуск 01/2004

A5E00261410-01

Предисловие, содержание	
Организационные блоки	1
Общие параметры для SFC	2
Функции копирования и манипулирования блоками	3
SFC для управления выполнением программы	4
SFC для управления системными часами	5
SFC для манипулирования счетчиками рабочего времени	6
SFC для передачи записей данных	7
DPV1 SFB стандарта PNO AK 1131	8
SFC для управления прерываниями по времени	9
SFC для управления прерываниями с задержкой	10
SFC для обработки синхронных ошибок	11
SFC для обработки прерываний и асинхронных ошибок	12
SFC для диагностики	13
SFC и SFB для обновления образа процесса и обработки битовых массивов	14
Системные функции для адресации модулей	15
SFC для децентрализованной периферии	16
SFC для связи с помощью глобальных данных	17
Общий обзор системы связи S7 и базовой системы связи S7	18
Система связи S7 (S7 Communication)	19
Коммуникационные SFC для неконфигурированных S7-соединений	20
PROFINet	21
Создание сообщений, связанных с блоками	22
Таймеры и счетчики IEC	23
Функции IEC	24
SFB с интегрированными функциями регулирования	25
SFB для компактных CPU	26
SFC для H CPU	27
Встроенные функции (для CPU со встроенными входами/выходами)	28
Гибкая технология	29
Диагностические данные	30
Списки состояний системы (SSL)	31
События	32
Списки SFC и SFB	33
Литература, глоссарий, предметный указатель	

## Указания по технике безопасности

Данное руководство содержит указания, которые вы должны соблюдать для обеспечения собственной безопасности, а также защиты от повреждений оборудования. Эти замечания выделены в руководстве символами, как показано ниже, в соответствии с уровнем опасности:



### Опасность

указывает, что несоблюдение надлежащих мер предосторожности может привести к гибели людей, тяжким телесным повреждениям или существенному имущественному ущербу.



### Предупреждение

указывает, что несоблюдение надлежащих мер предосторожности может привести к гибели людей, тяжким телесным повреждениям или к существенному имущественному ущербу.



### Предостережение

указывает, что несоблюдение надлежащих мер предосторожности может привести к легким телесным повреждениям и нанесению незначительного имущественного ущерба.

### Предостережение

указывает, что несоблюдение надлежащих мер предосторожности может привести к нанесению незначительного имущественного ущерба.

### Замечание

привлекает ваше внимание к особо важной информации о продукте, обращении с ним, или к соответствующей части документации.

## Квалифицированный персонал

К монтажу и работе на этом оборудовании должен допускаться только квалифицированный персонал. Квалифицированный персонал – это люди которые, имеют право вводить в эксплуатацию, заземлять и маркировать электрические цепи, оборудование и системы в соответствии с установленной практикой и стандартами техники безопасности.

## Надлежащее использование

Примите во внимание следующее:



### Предупреждение

Это устройство и его компоненты могут использоваться только для применений, описанных в каталоге или технической документации, и совместно только с теми устройствами или компонентами других производителей, которые были одобрены или рекомендованы фирмой Siemens.

Этот продукт может правильно и надежно функционировать только в том случае, если он правильно транспортируется, хранится, устанавливается и монтируется, а также если эксплуатируется и обслуживается в соответствии с рекомендациями.

## Товарные знаки

SIMATIC®, SIMATIC HMI® and SIMATIC NET® - это зарегистрированные товарные знаки SIEMENS AG.

Некоторые другие обозначения, использованные в этих документах, также являются зарегистрированными товарными знаками; если они используются третьей стороной для своих собственных целей, то соответствующие права собственности могут быть нарушены.

## Copyright © Siemens AG 2004 Все права защищены

Воспроизведение, передача или использование этого документа или его содержания не допускаются без письменного разрешения. Нарушители будут нести ответственность за нанесенный ущерб. Все права, включая права, вытекающие из предоставления патента или регистрации практической модели или конструкции, защищены.

## Отказ от ответственности

Мы проверили содержание этого руководства на соответствие с описанным аппаратным и программным обеспечением. Так как отклонения не могут быть полностью исключены, то мы не можем гарантировать полного соответствия. Однако данные, приведенные в этом руководстве, регулярно пересматриваются и все необходимые исправления вносятся в последующие издания. Мы будем благодарны за предложения по улучшению содержания.

# Предисловие

## Назначение

Это руководство дает полный обзор содержащихся в операционных системах CPU S7-300 и S7-400 организационных блоков (OB), системных функций (SFC), системных и стандартных функциональных блоков (SFB), а также функций IEC. В приложении описаны диагностические данные, списки состояний системы (SSL) и события.

---

### Примечание

Какие из этих функций и блоков в каких CPU доступны, Вы можете узнать из справочной части руководства "S7-300 Programmable Controller, Hardware and Installation [Программируемый контроллер S7-300. Аппаратные средства и монтаж]" [I70](#), или из справочного руководства "S7-400/M7-400 Programmable Controllers Module Specifications [Программируемые контроллеры S7-400/M7-400, Данные модулей]" [I101](#), или из "Instruction List: S7-400 Programmable Controller [Список команд: Программируемый контроллер S7-400]" [I102](#) (в зависимости от того, какая версия относится к Вашему CPU). Свойства CFB и функций передачи сигналов S7 Вы найдете в [I70](#) и [I101](#).

---

Информацию об операционных системах CPU, о разработке программ и о коммуникационных и диагностических возможностях CPU Вы получите в руководстве "Configuring Hardware and Communication Connections STEP 7 V5.1 [Конфигурирование аппаратуры и конфигурирование соединений с помощью STEP 7 V5.1]" [I234](#). Как вызвать функции и функциональные блоки в Вашей программе, объясняется в описаниях языков. Все названные функции программируются и параметрируются с помощью стандартного программного обеспечения STEP 7. Как использовать это программное обеспечение, описано в руководстве "Programming with STEP 7 V5.1 [Программирование с помощью STEP 7 версии 5.1]" [I231](#) и в оперативной помощи STEP 7.

## Круг читателей

Это руководство предназначено для программистов и инженеров, которые знакомы с процессами управления и которые отвечают за написание программ для программируемых логических контроллеров.

**Пакеты документации STEP 7**

Печатное издание 06/2000 является частью пакета "STEP 7 Basic Information [Базовая информация о STEP 7]".

В следующей таблице представлен обзор документации по STEP 7:

Документация	Назначение	Номер для заказа
Базовая информация о STEP 7, включающая в себя: <ul style="list-style-type: none"> <li>Working with STEP 7 V5.1, Getting Started Manual [Работа со STEP 7 версии 5.1. Введение в STEP 7]</li> <li>Programming with STEP 7 V5.1 [Программирование с помощью STEP 7 версии 5.1]</li> <li>Configuring Hardware and Communication Connections, STEP 7 V5.1 [Конфигурирование аппаратуры и конфигурирование соединений с помощью STEP 7 V5.1]</li> <li>From S5 to S7, Converter Manual [От S5 к S7. Руководство по конвертированию]</li> </ul>	Базовая информация для технического персонала, описывающая методы реализации задач управления с помощью STEP 7 и программируемых контроллеров S7-300/400.	6ES7810-4CA05-8BA0
Справочники по STEP 7, в том числе <ul style="list-style-type: none"> <li>Руководства Ladder Logic (LAD) /Function Block Diagram (FBD) /Statement List (STL) for S7-300/400 [Контактный план (LAD, KOP) /Функциональный план (FBD, FUP)/Список операторов (STL, AWL) для S7-300/400]</li> <li>Standard and System Functions for S7-300/400 [Стандартные и системные функции для S7-300/400]</li> </ul>	Предоставляется справочная информация и описываются языки программирования LAD (контактный план, KOP), FBD (функциональный план, FUP) и STL (список мнемоник, AWL) и стандартные и системные функции, т.е., расширенная базовая информация о STEP 7.	6ES7810-4CA05-8BR0

Интерактивная справочная система	Назначение	Номер для заказа
Помощь по STEP 7	Базовая информация о программировании и конфигурировании аппаратуры с помощью STEP 7 в виде оперативной справки (online).	Часть стандартного программного обеспечения STEP 7.
Справочная информация о STL/LAD/FBD Справочная информация о SFB/SFC Справочная информация об организационных блоках	Контекстно-чувствительная справочная информация	Часть стандартного программного обеспечения STEP 7.

## Интерактивная справочная система

Руководство дополняется интерактивной справочной системой, встроенной в программное обеспечение. Эта оперативная помощь направлена на то, чтобы снабдить Вас детальной поддержкой при использовании программного обеспечения.

Система помощи встроена в программное обеспечение с помощью ряда интерфейсов:

- Имеется несколько команд меню, которые Вы можете выбрать в меню **Help [Помощь]**:
  - Команда **Contents [Содержание]** открывает предметный указатель помощи по Step 7.
  - **Using Help [Использование помощи]** дает подробные инструкции по использованию оперативной помощи.
- Контекстная справочная система предлагает информацию о текущем контексте, например, об открытом диалоговом окне или об активном окне. Контекстно-чувствительную помощь можно открыть, щелкнув на кнопке "Help [Помощь]" или нажав F1.
- Строка состояния предлагает еще одну форму контекстно-чувствительной помощи. Она отображает краткое объяснение для каждой команды меню, когда указатель мыши расположен на этой команде.
- Краткое объяснение отображается также для каждой пиктограммы на панели инструментов, когда указатель мыши кратковременно помещается на этой пиктограмме.

Если Вы предпочитаете читать информацию, предоставляемую оперативной помощью, в печатном виде, то Вы можете распечатать отдельные темы помощи, книги или всю оперативную помощь.

Это руководство является извлечением из помощи по STEP 7, обновленной на гипертекстовом языке описания документов HTML. Так как данное руководство и оперативная помощь имеют почти одинаковую структуру, то не составляет труда переключаться между руководством и оперативной помощью.

## Другие руководства

Различные CPU S7-300 и S7-400, а также модули S7-300 и S7-400 описаны в следующих руководствах:

- для программируемых логических контроллеров S7-300: „S7-300 Programmable Controller, Hardware and Installation [Программируемый контроллер S7-300. Аппаратные средства и монтаж]“ [/170/](#), „S7-300, M7-300 Programmable Controllers Module Specifications [Спецификации модулей программируемых логических контроллеров S7-300, M7-300]“ [/71/](#) и в списке команд [/72/](#).
- для программируемых логических контроллеров S7-400: „S7-400/M7-400 Programmable Controllers Module Specifications [Программируемые контроллеры S7-400/M7-400, Данные модулей]“ [/101/](#) и в списке команд [/102/](#).

## Как пользоваться этим руководством

Это руководство включает в себя следующие темы:

- В главе 1 объясняются функции всех организационных блоков.
- Глава 2 описывает общие параметры RET\_VAL, REQ и BUSY.
- Главы с 3 по 28 описывают SFC, SFB и функции IEC.
- Главы с 29 по 32 содержат описание структуры диагностических данных, обзор идентификаторов списков состояний системы (SSL-ID), возможные события, списки SFC, SFB и FC, описанных в этом руководстве, обзор SDB и список литературы, использованной в этом руководстве.
- В разделе библиографии представлен список руководств.
- В глоссарии объяснены важные термины.
- Предметный указатель поможет Вам быстро найти нужные разделы текста и темы.

## Соглашения

Ссылки на другие руководства и документацию даются с помощью номеров, заключенных между косыми чертами /.../. Эти номера относятся к названиям руководств, перечисленных в библиографии.

## Специальное указание

Системные функции могут прерываться. Если имеются какие-либо ограничения, относящиеся к определенным SFC или ситуациям, то они объясняются в описаниях конкретных SFC.

## Учебный центр

Для того чтобы помочь Вам познакомиться с работой с S7 PLC, мы организовали ряд учебных курсов. Обратитесь в наш региональный учебный центр или в центральный учебный центр в Нюрнберге D 90327.

Тлф: +49 (911) 895-3200.

Интернет: <http://www.sitrain.com>

## Техническая поддержка A&D (A&D Technical Support)

Круглосуточно, по всему миру:



<p><b>Всемирная (Нюрнберг) техническая поддержка</b></p> <p>Мест. вр. : 0:00 - 24:00 / 365 дней Тлф: +49 (0) 180 5050-222 Факс: +49 (0) 180 5050-223 E-mail: adsupport@siemens.com GMT: +1:00</p>		
<p><b>Европа / Африка (Нюрнберг) Авторизация</b></p> <p>Мест. вр. : Пнд.-Птн. 8:00 - 17:00 Тлф: +49 (0) 180 5050-222 Факс: +49 (0) 180 5050-223 E-mail: adautorisierung@siemens.com GMT: +1:00</p>	<p><b>США (Джонсон-Сити) Техническая поддержка и авторизация</b></p> <p>Мест. вр. : Пнд.-Птн. 8:00 - 17:00 Тлф: +1 (0) 423 262 2522 Факс: +1 (0) 423 262 2289 E-mail: simatic.hotline@sea.siemens.com GMT: -5:00</p>	<p><b>Азия / Австралия (Пекин) Техническая поддержка и авторизация</b></p> <p>Мест. вр. : Пнд.-Птн. 8:30 - 17:30 Тлф: +86 10 64 75 75 75 Факс: +86 10 64 74 74 74 E-mail: adsupport.asia@siemens.com GMT: +8:00</p>
<p>Язык общения для технической поддержки и авторизации: английский или немецкий.</p>		

## **Обслуживание и поддержка с помощью Интернета**

В дополнение к документации, Вы можете также получить нашу поддержку с помощью Интернета:

<http://www.siemens.com/automation/service&support>

Здесь Вы найдете:

- Новую информацию, которая постоянно обновляется, с последними данными по продуктам, которые Вы используете.
- Правовую документацию, необходимую Вам для поиска в службе обслуживания и поддержки с помощью Интернета (Service & Support).
- Форум для обмена информацией и опытом между специалистами и пользователями.
- Контактные реквизиты для ответов на вопросы по автоматизации и приводам (Automation & Drives) в нашей базе данных.
- Информацию по местным отделениям обслуживания, по ремонту и запчастям, а также многое другое под заголовком "Services" ("Обслуживание").



# Содержание

<b>1</b>	<b>Организационные блоки</b>	<b>1-1</b>
1.1	Обзор организационных блоков (ОВ)	1-2
1.2	Организационный блок циклического выполнения программы (ОВ1)	1-5
1.3	Организационные блоки прерываний по времени (ОВ10 – ОВ17)	1-7
1.4	Организационные блоки прерываний с задержкой (ОВ20 – ОВ23)	1-11
1.5	Организационные блоки циклических прерываний (ОВ30 – ОВ38)	1-13
1.6	Организационные блоки аппаратных прерываний (ОВ40 – ОВ47)	1-15
1.7	Организационный блок прерывания статуса (ОВ55)	1-17
1.8	Организационный блок прерывания модификации (ОВ56)	1-18
1.9	Организационный блок прерывания, определяемого производителем (ОВ57)	1-19
1.10	Организационный блок мультипроцессорных прерываний (ОВ60)	1-20
1.11	Организационный блок синхронного циклического прерывания (ОВ61)	1-22
1.12	ОВ ошибок резервирования входов/выходов (ОВ70)	1-23
1.13	ОВ ошибок резервирования CPU (ОВ72)	1-25
1.14	ОВ ошибок резервирования соединений (ОВ73)	1-27
1.15	Организационный блок ошибок времени (ОВ80)	1-29
1.16	Организационный блок неисправностей источника питания (ОВ81)	1-31
1.17	Организационный блок диагностических прерываний (ОВ82)	1-33
1.18	Организационный блок снятия/установки модулей (ОВ83)	1-35
1.19	Организационный блок аппаратных ошибок CPU (ОВ84)	1-37
1.20	Организационный блок ошибок класса приоритета (ОВ85)	1-38
1.21	Организационный блок неисправностей стоек (ОВ86)	1-41
1.22	Организационный блок коммуникационных ошибок (ОВ87)	1-44
1.23	ОВ обработки прерывания выполнения программы (ОВ 88)	1-46
1.24	Организационный блок фонового режима (ОВ90)	1-47
1.25	Организационные блоки запуска (ОВ100, ОВ101 и ОВ102)	1-49
1.26	Организационный блок ошибок программирования (ОВ121)	1-53
1.27	Организационный блок ошибок доступа к периферии (ОВ122)	1-56
<b>2</b>	<b>Общие параметры для SFC</b>	<b>2-1</b>
2.1	Оценка ошибок с помощью выходного параметра RET_VAL	2-2
2.2	Значение параметров REQ, RET_VAL и BUSY для асинхронных SFC	2-6

<b>3</b>	<b>Функции копирования и манипулирования блоками</b>	<b>3-1</b>
3.1	Копирование переменных с помощью SFC20 "BLKMOV"	3-2
3.2	Непрерываемое копирование переменных с помощью SFC81 "UBLKMOV"	3-5
3.3	Инициализация области памяти с помощью SFC21 "FILL"	3-7
3.4	Создание блока данных с помощью SFC22 "CREAT_DB"	3-9
3.5	Удаление блока данных с помощью SFC23 "DEL_DB"	3-11
3.6	Тестирование блока данных с помощью SFC24 "TEST_DB"	3-13
3.7	Сжатие памяти пользователя с помощью SFC25 "COMPRESS"	3-14
3.8	Передача заменяющего значения в аккумулятор 1 с помощью SFC44 "REPL_VAL"	3-16
3.9	Генерация блоков данных в загружаемой памяти с помощью SFC82 "CREA_DBL"	3-17
3.10	Считывание информации из блока данных в загружаемой памяти с помощью SFC83 "READ_DBL"	3-20
3.11	Запись блока данных в загружаемой памяти с помощью SFC84 "WRITE_DBL"	3-22
3.12	Создание блока данных с помощью SFC85 "CREA_DB"	3-24
<b>4</b>	<b>SFC для управления выполнением программы</b>	<b>4-1</b>
4.1	Повторный запуск контроля времени цикла с помощью SFC43 "RE_TRIGR"	4-2
4.2	Перевод CPU в STOP с помощью SFC46 "STP"	4-3
4.3	Задержка исполнения программы пользователя с помощью SFC47 "WAIT"	4-4
4.4	Запуск прерывания многопроцессорной обработки с помощью SFC35 "MP_ALM"	4-5
4.5	Управление функцией CiR с помощью SFC104 "CiR"	4-6
<b>5</b>	<b>SFC для управления системными часами</b>	<b>5-1</b>
5.1	Установка времени с помощью SFC0 "SET_CLK"	5-2
5.2	Считывание времени с помощью SFC1 "READ_CLK"	5-3
5.3	Синхронизация ведомых часов с помощью SFC48 "SNC_RTCB"	5-4
5.4	Установка времени суток и TOD-статуса с помощью SFC100 "SET_CLKS"	5-5
<b>6</b>	<b>SFC для манипулирования счетчиками рабочего времени</b>	<b>6-1</b>
6.1	Счетчики рабочего времени	6-2
6.2	Управление счетчиками рабочего времени с помощью SFC101 "RTM"	6-3
6.3	Установка счетчика рабочего времени с помощью SFC2 "SET_RTM"	6-5
6.4	Запуск и останов счетчика рабочего времени с помощью SFC3 "CTRL_RTM"	6-6
6.5	Считывание значения счетчика рабочего времени с помощью SFC4 "READ_RTM"	6-7
6.6	Считывание системного времени с помощью SFC64 "TIME_TCK"	6-8
<b>7</b>	<b>SFC для передачи записей данных</b>	<b>7-1</b>
7.1	Запись и чтение записей данных	7-2

7.2	Чтение определенных параметров с помощью SFC54 "RD_DPARM"	7-4
7.3	Чтение заранее определенных параметров с помощью SFC102 "RD_DPARA"	7-5
7.4	Запись динамических параметров с помощью SFC55 "WR_PARM"	7-6
7.5	Запись параметров, установленных по умолчанию, с помощью SFC56 "WR_DPARM"	7-8
7.6	Назначение параметров для модуля с помощью SFC57 "PARM_MOD"	7-9
7.7	Внесение записи данных с помощью SFC58 "WR_REC"	7-12
7.8	Чтение записи данных с помощью SFC59 "RD_REC"	7-13
7.9	Чтение записи данных с помощью SFC59 "RD_REC" в CPU S7-300	7-19
7.10	Дополнительная информация об ошибках SFC 55–59	7-22
<b>8</b>	<b>DPV1 SFB стандарта PNO АК 1131</b>	<b>8-1</b>
8.1	Чтение записи данных из ведомого DP-устройства с помощью SFB52 "RDREC"	8-2
8.2	Внесение записи данных в ведомый DP-компонент с помощью SFB53 "WRREC"	8-4
8.3	Получение прерывания от ведомого DP-устройства посредством SFB54 "RALRM"	8-6
<b>9</b>	<b>SFC для управления прерываниями по времени</b>	<b>9-1</b>
9.1	Манипулирование прерываниями по времени	9-2
9.2	Характеристики SFC 28 ... 31	9-3
9.3	Установка прерывания по времени с помощью SFC28 "SET_TINT"	9-5
9.4	Отмена прерывания по времени с помощью SFC29 "CAN_TINT"	9-6
9.5	Активация прерывания по времени с помощью SFC30 "ACT_TINT"	9-7
9.6	Опрос прерывания по времени с помощью SFC31 "QRY_TINT"	9-8
<b>10</b>	<b>SFC для управления прерываниями с задержкой</b>	<b>10-1</b>
10.1	Манипулирование прерываниями с задержкой	10-2
10.2	Запуск прерывания с задержкой с помощью SFC32 "SRT_DINT"	10-4
10.3	Опрос состояния прерывания с задержкой с помощью SFC34 "QRY_DINT"	10-5
10.4	Отмена прерывания с задержкой с помощью SFC33 "CAN_DINT"	10-6
<b>11</b>	<b>SFC для обработки синхронных ошибок</b>	<b>11-1</b>
11.1	Маскирование синхронных ошибок	11-2
11.2	Маскирование синхронных ошибок с помощью SFC36 "MSK_FLT"	11-12
11.3	Демаскирование синхронных ошибок с помощью SFC37 "DMSK_FLT"	11-13
11.4	Чтение регистра ошибок с помощью SFC38 "READ_ERR"	11-14
<b>12</b>	<b>SFC для обработки прерываний и асинхронных ошибок</b>	<b>12-1</b>
12.1	Задержка и блокировка прерываний и асинхронных ошибок	12-2
12.2	Блокировка обработки новых прерываний и асинхронных ошибок с помощью SFC39 "DIS_IRT"	12-4
12.3	Разблокировка обработки новых прерываний и асинхронных ошибок	12-6

	с помощью SFC40 "EN_IRT"	
12.4	Задержка обработки прерываний более высокого приоритета и асинхронных ошибок с помощью SFC41 "DIS_AIRT"	12-7
12.5	Разблокировка обработки прерываний более высокого приоритета и асинхронных ошибок с помощью SFC42 "EN_AIRT"	12-8
<b>13</b>	<b>SFC для диагностики</b>	<b>13-1</b>
13.1	Системная диагностика	13-2
13.2	Чтение стартовой информации ОВ с помощью SFC6 "RD_SINFO"	13-3
13.3	Чтение списка состояний системы или подписка с помощью SFC51 "RDSYSST"	13-6
13.4	Запись диагностического события, определенного пользователем, в диагностический буфер с помощью SFC52 "WR_USMSG"	13-11
13.5	Определение времени выполнения программы ОВ с помощью SFC78 "OB_RT"	13-16
13.6	Запрос данных фактического состояния соединений с помощью SFC87 "C_DIAG"	13-20
13.7	Идентификация топологии шины системы ведущего DP-устройства с помощью SFC103 "DP_TOPOL"	13-24
<b>14</b>	<b>SFC и SFB для обновления образа процесса и обработки битовых массивов</b>	<b>14-1</b>
14.1	Обновление таблицы входов образа процесса с помощью SFC26 "UPDAT_PI"	14-2
14.2	Обновление таблицы выходов образа процесса с помощью SFC27 "UPDAT_PO"	14-4
14.3	Установка области выходов с помощью SFC79 "SET"	14-5
14.4	Обновление таблицы входов раздела образа процесса в синхронном цикле с помощью SFC 126 "SYNC_PI"	14-6
14.5	Обновление таблицы выходов раздела образа процесса в синхронном цикле с помощью SFC 127 "SYNC_PO"	14-8
14.6	Сброс области выходов с помощью SFC80 "RSET"	14-10
14.7	Реализация генератора последовательностей с помощью SFB32 "DRUM"	14-11
<b>15</b>	<b>Системные функции для адресации модулей</b>	<b>15-1</b>
15.1	Определение логического базового адреса модуля с помощью SFC5 "GADR_LGC"	15-2
15.2	Определение слота модуля, соответствующего логическому адресу, с помощью SFC49 "LGC_GADR"	15-4
15.3	Определение всех логических адресов модуля с помощью SFC50 "RD_LGADR"	15-6
<b>16</b>	<b>SFC для децентрализованной периферии</b>	<b>16-1</b>
16.1	Запуск аппаратного прерывания в ведущем DP-устройстве с помощью SFC7 "DP_PRAL"	16-2
16.2	Синхронизация групп ведомых DP-устройств с помощью SFC11 "DPSYC_FR"	16-5
16.3	Активация и деактивация ведомых (slave) DP-устройств с помощью SFC12 "D_ACT_DP"	16-11

16.4	Чтение диагностических данных ведомые DP-устройства (диагностика slave-устройства) с помощью SFC 13 "DPNRM_DG"	16-16
16.5	Чтение консистентных данных стандартного ведомые DP-устройства с помощью SFC14 "DPRD_DAT"	16-19
16.6	Запись консистентных данных в стандартное ведомое DP-устройство с помощью SFC15 "DPWR_DAT"	16-21
<b>17</b>	<b>SFC для связи с помощью глобальных данных</b>	<b>17-1</b>
17.1	Передача GD-пакета с помощью SFC60 "GD_SND"	17-2
17.2	Извлечение принятого GD-пакета с помощью SFC61 "GD_RCV"	17-4
<b>18</b>	<b>Общий обзор системы связи S7 и базовой системы связи S7</b>	<b>18-1</b>
18.1	Различие между блоками системы связи S7 (S7 Communication) и базовой системы связи S7 (S7 Basic Communication)	18-2
18.2	Консистентность данных	18-5
18.3	Краткий обзор коммуникационных блоков S7	18-7
18.4	Краткий обзор блоков для базовой системы связи S7 (S7 Basic Communication)	18-11
<b>19</b>	<b>Система связи S7 (S7 Communication)</b>	<b>19-1</b>
19.1	Общие параметры блоков SFB/FB и функций SFC/FC системы связи S7 (S7 Communication)	19-2
19.2	Подпрограмма запуска SFB для сконфигурированного S7-соединения	19-7
19.3	Реакция блоков SFB на различные ситуации в системе	19-10
19.4	Некоординированная передача данных посредством SFB 8/FB 8 "USEND"	19-12
19.5	Некоординированный прием данных с помощью SFB9/ FB9 "URCV"	19-15
19.6	Передача сегментированных данных с помощью SFB/FB12 "BSEND"	19-18
19.7	Прием сегментированных данных с помощью SFB/FB13 "BRCV"	19-23
19.8	Запись данных в удаленный CPU с помощью SFB/FB15 "PUT"	19-27
19.9	Чтение данных из удаленного CPU с помощью SFB/FB14 "GET"	19-30
19.10	Передача данных на принтер с помощью SFB16 "PRINT"	19-33
19.11	Инициализация теплого или холодного рестарта в удаленном устройстве с помощью SFB19 "START"	19-39
19.12	Переключение удаленного устройства в состояние STOP с помощью SFB20 "STOP"	19-42
19.13	Инициализация горячего рестарта в удаленном устройстве с помощью SFB21 "RESUME"	19-44
19.14	Запрос состояния удаленного партнера с помощью SFB22 "STATUS"	19-46
19.15	Прием состояния удаленного устройства с помощью SFB23 "USTATUS"	19-48
19.16	Запрос состояния соединения, относящегося к экземпляру коммуникационного SFB, с помощью SFC62 "CONTROL"	19-51
19.17	Запрос состояния соединения с помощью FC62 "C_CNTRL"	19-54
19.18	Требования к рабочей памяти для SFB/FB системы связи S7 (S7 Communication)	19-56
<b>20</b>	<b>Коммуникационные SFC для несконфигурированных S7-соединений</b>	<b>20-1</b>

20.1	Общие параметры коммуникационных SFC	20-2
20.2	Информация об ошибках коммуникационных SFC для неконфигурированных S7-соединений	20-4
20.3	Передача данных партнеру по связи, находящемуся вне локальной станции S7, с помощью SFC65 "X_SEND"	20-7
20.4	Прием данных от партнера по связи, находящегося вне локальной станции S7, с помощью SFC66 "X_RCV"	20-9
20.5	Запись данных в партнере по связи вне локальной станции S7 с помощью SFC68 "X_PUT"	20-13
20.6	Чтение данных из партнера по связи, находящегося вне локальной станции S7, с помощью SFC67 "X_GET"	20-15
20.7	Прерывание существующего соединения с партнером по связи, находящимся вне локальной станции S7, с помощью SFC69 "X_ABORT"	20-17
20.8	Чтение данных из партнера по связи, находящегося в пределах локальной станции S7, с помощью SFC72 "I_GET"	20-18
20.9	Запись данных в партнере по связи, находящемся в пределах локальной станции S7, с помощью SFC73 "I_PUT"	20-20
20.10	Прерывание существующего соединения с партнером по связи, находящимся в пределах локальной станции S7, с помощью SFC74 "I_ABORT"	20-22
<b>21</b>	<b>PROFINet</b>	<b>21-1</b>
21.1	Основные сведения, касающиеся SFC 112, 113 и 114	21-2
21.2	Обновление входов UPI для компонентов PROFINet с помощью SFC112 "PN_IN"	21-5
21.3	Обновление выходов UPI для компонентов PROFINet с помощью SFC113 "PN_OUT"	21-6
21.4	Обновление соединений в системе распределенной периферии (DP interconnections) с помощью SFC114 "PN_DP"	21-7
<b>22</b>	<b>Создание сообщений, связанных с блоками</b>	<b>22-1</b>
22.1	Введение в создание сообщений, связанных с блоками, с помощью SFB	22-2
22.2	Создание с помощью SFB36 "NOTIFY" сообщений, связанных с блоками, без квитирования	22-7
22.3	Создание с помощью SFB31 "NOTIFY_8P" сообщений, связанных с блоками, без отображения квитирования	22-9
22.4	Создание с помощью SFB33 "ALARM" сообщений, связанных с блоками, с квитированием	22-12
22.5	Создание сообщений, связанных с блоками, с сопутствующими значениями для восьми сигналов с помощью SFB35 "ALARM_8P"	22-15
22.6	Создание сообщений, связанных с блоками, без сопутствующих значений для восьми сигналов с помощью SFB34 "ALARM_8"	22-18
22.7	Передача архивных данных с помощью SFB37 "AR_SEND"	22-20
22.8	Блокировка сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии с помощью SFC10 "DIS_MSG"	22-22
22.9	Разблокировка сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии с	22-24

	помощью SFC9 "EN_MSG"	
22.10	Поведение SFB для создания сообщений, связанных с блоками, при пуске	22-26
22.11	Как SFB для создания сообщений, связанных с блоками, реагируют на неисправности	22-27
22.12	Введение в создание сообщений, связанных с блоками, с помощью SFC	22-28
22.13	Создание квитируемых сообщений, связанных с блоками, с помощью SFC 17 "ALARM_SQ" и всегда квитируемых сообщений, связанных с блоками, с помощью SFC18 "ALARM_S"	22-31
22.14	Определение состояния квитирувания последнего сообщения для наступающего события ALARM_SQ/ ALARM_DQ с помощью SFC19 "ALARM_SC"	22-34
22.15	Создание квитируемых сообщений, связанных с блоками, с помощью SFC 107 "ALARM_DQ" и всегда квитируемых сообщений, связанных с блоками, с помощью SFC108 "ALARM_D"	22-35
22.16	Считывание данных из динамически занимаемых областей системной памяти с помощью SFC105 "READ_SI"	22-38
22.17	Удаление данных из динамически занимаемых областей системной памяти с помощью SFC106 "DEL_SI"	22-41
<b>23</b>	<b>Таймеры и счетчики IEC</b>	<b>23-1</b>
23.1	Генерирование импульса с помощью SFB3 "TP"	23-2
23.2	Генерирование задержки включения с помощью SFB4 "TON"	23-4
23.3	Генерирование задержки выключения с помощью SFB5 "TOF"	23-6
23.4	Прямой счет с помощью SFB0 "CTU"	23-8
23.5	Обратный счет с помощью SFB1 "CTD"	23-9
23.6	Прямой и обратный счет с помощью SFB2 "CTUD"	23-10
<b>24</b>	<b>Функции IEC</b>	<b>24-1</b>
24.1	Обзор	24-2
24.2	Технические данные функций IEC	24-3
24.3	Дата и время как составные типы данных	24-4
24.4	Функции даты и времени суток	24-5
24.5	Сравнение переменных типа DATE_AND_TIME	24-9
24.6	Сравнение переменных типа STRING	24-11
24.7	Редактирование числовых значений	24-14
24.8	Пример на STL	24-15
24.9	Пример на STL	24-16
24.10	Редактирование переменных типа STRING	24-17
24.11	Преобразование форматов типов данных	24-22
<b>25</b>	<b>SFB с интегрированными функциями регулирования</b>	<b>25-1</b>
25.1	Непрерывное регулирование с помощью SFB 41 /FB 41 "CONT_C"	25-2
25.2	Пошаговое регулирование с помощью SFB42 / FB42 "CONT_S"	25-10
25.3	Формирование импульсов с помощью SFB43 / FB43 "PULSEGEN"	25-17

25.4	Пример блока PULSEGEN	25-28
<b>26</b>	<b>SFB для компактных CPU</b>	<b>26-1</b>
26.1	Позиционирование с аналоговым выходом с использованием SFB 44 "Analog"	26-2
26.2	Позиционирование для дискретного выхода с использованием SFB 46 "DIGITAL"	26-18
26.3	Управление счетчиком с использованием SFB 47 "COUNT"	26-34
26.4	Управление измерением частоты с использованием SFB 48 "FREQUENCY"	26-40
26.5	Управление ШИМ-модуляцией с использованием SFB 49 "PULSE"	26-45
26.6	Пересылка данных (ASCII, 3964(R)) с использованием SFB 60 "SEND_PTP"	26-49
26.7	Прием данных (ASCII, 3964(R)) с использованием SFB 61 "RCV_PTP"	26-53
26.8	Очистка буфера приема (ASCII, 3964(R)) с использованием SFB 62 "RES_RCVB"	26-57
26.9	Пересылка данных (512(R)) с использованием SFB 63 "SEND_RK"	26-60
26.10	Выборка данных (512(R)) с использованием SFB 64 "FETCH RK"	26-65
26.11	Прием и выдача данных (512(R)) с использованием SFB 65 "SERVE_RK"	26-71
26.12	Дополнительная информация об ошибках для SFB 60...65	26-77
<b>27</b>	<b>SFC для H CPU</b>	<b>27-1</b>
27.1	Управление функционированием в H-системах с помощью SFC90 "H_CTRL"	27-2
<b>28</b>	<b>Встроенные функции (для CPU со встроенными входами/выходами)</b>	<b>28-1</b>
28.1	SFB29 (HS_COUNT)	28-2
28.2	SFB30 (FREQ_MES)	28-4
28.3	SFB38 (HSC_A_B)	28-5
28.4	SFB39 (POS)	28-6
<b>29</b>	<b>Гибкая технология</b>	<b>29-1</b>
29.1	SFC63 (AB_CALL)	29-2
<b>30</b>	<b>Диагностические данные</b>	<b>30-1</b>
30.1	Обзор структуры диагностических данных	30-2
30.2	Диагностические данные	30-3
30.3	Структура диагностических данных, относящихся к каналу	30-5
<b>31</b>	<b>Списки состояний системы (SSL)</b>	<b>31-1</b>
31.1	Обзор списков состояний системы (SSL)	31-2
31.2	Структура подписка SSL	31-3
31.3	SSL-ID	31-4



31.4	Возможные подписки состояний системы	31-5
31.5	SSL-ID W#16#ху11 – Идентификация модуля	31-6
31.6	SSL-ID W#16#ху12 – Характеристики CPU	31-7
31.7	SSL-ID W#16#ху13 – Области памяти	31-9
31.8	SSL-ID W#16#ху14 – Системные области	31-10
31.9	SSL-ID W#16#ху15 – Типы блоков	31-12
31.10	SSL-ID W#16#ху19 – Состояние светодиодов модуля	31-13
31.11	SSL-ID W#16#ху1С – Идентификация компонентов	31-15
31.12	SSL-ID W#16#ху22 – Состояние прерывания	31-17
31.13	SSL-ID W#16#ху25 - Назначение разделов образа процесса для ОВ	31-19
31.14	SSL-ID W#16#ху32 – Данные о состоянии связи	31-22
31.15	Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#0005	31-23
31.16	Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#0008	31-24
31.17	Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#000В	31-26
31.18	Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#000С	31-27
31.19	Запись данных фрагмента подписка с SSL-ID W#16#0232 и индексом W#16#0004	31-28
31.20	SSL-ID W#16#ху71 – Групповая информация Н CPU	31-29
31.21	SSL-ID W#16#ху74 – Состояние светодиодов модуля	31-32
31.22	SSL-ID W#16#ху75 – Состояние связи между Н-системой и подключенными ведомыми DP-устройствами	31-34
31.23	SSL-ID W#16#ху90 – состояние связи для всех ведущих DP-систем, зарегистрированных в CPU	31-36
31.24	SSL-ID W#16#ху91 – Информация о состоянии модуля	31-38
31.25	SSL-ID W#16#ху92 – Информация о состоянии стойки/станции	31-44
31.26	SSL-ID W#16#ху95 - расширенная информация о состоянии систем ведущих DP-устройств	31-48
31.27	SSL-ID W#16#хуA0 – Диагностический буфер	31-50
31.28	SSL-ID W#16#00В1 – Диагностическая информация модуля	31-51
31.29	SSL-ID W#16#00В2 – Диагностическая запись данных 1 с географическим адресом	31-53
31.30	SSL-ID W#16#00В3 – Диагностические данные модуля с логическим базовым адресом	31-54
31.31	SSL-ID W#16#00В4 – Диагностические данные ведомых DP-устройств	31-55
<b>32</b>	<b>События</b>	<b>32-1</b>
32.1	События и идентификаторы (ID) событий	32-2
32.2	Класс событий 1 – События, связанные со стандартными ОВ	32-4
32.3	Класс событий 2 – Синхронные ошибки	32-5
32.4	Класс событий 3 – Асинхронные ошибки	32-6
32.5	Класс событий 4 – События, связанные с режимом STOP, и другие изменения режима работы	32-9
32.6	Класс событий 5 – События этапа выполнения	32-13

32.7	Класс событий 6 – Коммуникационные события	32-14
32.8	Класс событий 7 - События H/F	32-16
32.9	Класс событий 8 – Диагностические события для модулей	32-18
32.10	Класс событий 9 – Стандартные события пользователя	32-20
32.11	Классы событий А и В – Свободные пользовательские события	32-22
32.12	Резервные классы событий	32-23
<b>33</b>	<b>Списки SFC и SFB</b>	<b>33-1</b>
33.1	Список SFC, упорядоченный по номерам	33-2
33.2	Список SFC, упорядоченный по алфавиту	33-5
33.3	Список SFC, упорядоченный по алфавиту	33-8
33.4	Список SFB, упорядоченный по алфавиту	33-12
<b>Литература</b>		<b>L-1</b>
<b>Глоссарий</b>		<b>G-1</b>
<b>Предметный указатель</b>		<b>I-1</b>

# 1 Организационные блоки

## 1.1 Обзор организационных блоков (ОВ)

### Что такое организационные блоки?

Организационные блоки образуют интерфейс между операционной системой CPU и программой пользователя. ОВ используются для исполнения определенных разделов программы:

- при запуске CPU
- при циклическом или зависящем от времени исполнении программы
- при возникновении ошибок
- при возникновении аппаратных прерываний.

Организационные блоки исполняются в соответствии с присвоенными им приоритетами.

### Какие ОВ имеются в распоряжении?

Не все CPU могут обрабатывать все ОВ, доступные в STEP 7. Обратитесь к техническому описанию своего CPU, чтобы определить, какие ОВ могут использоваться Вашим CPU.

### Где найти дополнительную информацию?

Для получения дополнительной информации обращайтесь к оперативной помощи в режиме online или к следующим руководствам:

- **/70/**: Это руководство содержит справочные данные о функциональных возможностях различных CPU S7-300. Там же описаны возможные стартовые события для каждого ОВ.
- **/101/**: Это руководство содержит справочные данные о функциональных возможностях различных CPU S7-400. Там же описаны возможные стартовые события для каждого ОВ.

Следующая таблица содержит стартовое событие и установленный по умолчанию класс приоритета для каждого ОВ.

ОВ	Стартовое событие	Класс приоритета по умолчанию	Объяснение
ОВ1	Конец запуска или конец ОВ1	1	Свободный цикл
ОВ10	Прерывание по времени 0	2	Время по умолчанию не определено
ОВ11	Прерывание по времени 1	2	
ОВ12	Прерывание по времени 2	2	
ОВ13	Прерывание по времени 3	2	
ОВ14	Прерывание по времени 4	2	
ОВ15	Прерывание по времени 5	2	
ОВ16	Прерывание по времени 6	2	
ОВ17	Прерывание по времени 7	2	
ОВ20	Прерывание с задержкой 0	3	Время по умолчанию не определено
ОВ21	Прерывание с задержкой 1	4	
ОВ22	Прерывание с задержкой 2	5	
ОВ23	Прерывание с задержкой 3	6	
ОВ30	Циклическое прерывание 0 (интервал по умолчанию: 5 с)	7	Циклические прерывания
ОВ31	Циклическое прерывание 1 (интервал по умолчанию: 2 с)	8	
ОВ32	Циклическое прерывание 2 (интервал по умолчанию: 1 с)	9	
ОВ33	Циклическое прерывание 3 (интервал по умолчанию: 500 мс)	10	
ОВ34	Циклическое прерывание 4 (интервал по умолчанию: 200 мс)	11	
ОВ35	Циклическое прерывание 5 (интервал по умолчанию: 100 мс)	12	
ОВ36	Циклическое прерывание 6 (интервал по умолчанию: 50 мс)	13	
ОВ37	Циклическое прерывание 7 (интервал по умолчанию: 20 мс)	14	
ОВ38	Циклическое прерывание 8 (интервал по умолчанию: 10 мс)	15	
ОВ40	Аппаратное прерывание 0	16	Аппаратные прерывания
ОВ41	Аппаратное прерывание 1	17	
ОВ42	Аппаратное прерывание 2	18	
ОВ43	Аппаратное прерывание 3	19	
ОВ44	Аппаратное прерывание 4	20	
ОВ45	Аппаратное прерывание 5	21	
ОВ46	Аппаратное прерывание 6	22	
ОВ47	Аппаратное прерывание 7	23	
ОВ55	Прерывание состояния (статуса)	2	Прерывания DPV1
ОВ56	Прерывание модификации параметров (update)	2	
ОВ57	Прерывание, определяемое производителем	2	
ОВ60	Вызов SFC35 "MP_ALM"	25	Прерывание мультипроцессорного режима
ОВ61	Синхронное циклическое прерывание 1	25	Синхронное циклическое прерывание
ОВ62	Синхронное циклическое прерывание 2	25	
ОВ63	Синхронное циклическое прерывание 3	25	
ОВ64	Синхронное циклическое прерывание 4	25	

OB70	Ошибка резервирования ввода/вывода	25	Прерывания по ошибке резервирования
OB72	Ошибка резервирования CPU	28	
OB73	Ошибка резервирования коммуникаций (только для H CPU)	25	
OB80	Ошибка времени	26, 28 <sup>1)</sup>	Прерывания в результате асинхронной ошибки
OB81	Неисправность источника питания	26, 28 <sup>1)</sup>	
OB82	Диагностическое прерывание	26, 28 <sup>1)</sup>	
OB83	Прерывание установки/снятия модуля	26, 28 <sup>1)</sup>	
OB84	Аппаратная ошибка CPU	26, 28 <sup>1)</sup>	
OB85	Ошибка исполнения программы	26, 28 <sup>1)</sup>	
OB86	Выход из строя стойки расширения, master-системы DP или станции для децентрализованной периферии	26, 28 <sup>1)</sup>	
OB87	Коммуникационная ошибка	26, 28 <sup>1)</sup>	
OB88	Прерывание обработки	28	
OB90	Теплый или холодный рестарт, или удаление блока, исполняемого в OB90, или загрузка OB90 в CPU, или завершение OB90	29 <sup>2)</sup>	Фоновый цикл
OB100	Теплый рестарт	27 <sup>1)</sup>	Запуск
OB101	Горячий рестарт	27 <sup>1)</sup>	
OB102	Холодный рестарт	27 <sup>1)</sup>	
OB121	Ошибка программирования	Приоритет OB, вызвавшего ошибку	Прерывания в результате синхронной ошибки
OB122	Ошибка доступа к периферии	Приоритет OB, вызвавшего ошибку	

<sup>1)</sup> Классы приоритетов 27 и 28 действительны в модели классов приоритетов запуска.

<sup>2)</sup> Класс приоритета 29 соответствует приоритету 0.29. Фоновый цикл, таким образом, имеет более низкий приоритет, чем свободный цикл.

## 1.2 Организационный блок циклического выполнения программы (OB1)

### Описание

Операционная система CPU S7 исполняет OB1 непрерывно. Когда OB1 исполнен, операционная система начинает его обработку вновь. Циклическая обработка OB начинается по окончании стадии запуска. Вы можете вызывать в OB1 функциональные блоки (FB, SFB) или функции (FC, SFC).

### Принцип действия OB1

OB1 имеет самый низкий приоритет среди всех OB, время выполнения которых контролируется, иными словами, все остальные OB, кроме OB90, могут прерывать выполнение OB1. Операционная система вызывает OB1 при следующих событиях:

- Завершение запуска.
- Конец обработки OB 1 (предыдущего цикла).

Когда OB1 исполнен, операционная система отправляет глобальные данные. Перед повторным запуском OB1 операционная система записывает таблицу выходов образа процесса в модули вывода, обновляет таблицу входов образа процесса и получает глобальные данные для CPU.

S7 осуществляет контроль максимальной длительности цикла сканирования, чем гарантируется максимальное время реакции. Значение максимальной длительности цикла сканирования установлено по умолчанию на 150 мс. Вы можете установить новое значение или перезапустить контроль времени в любой точке Вашей программы при помощи SFC43 "RE\_TRIGR". Если время выполнения программы превышает максимальное время цикла OB 1, то операционная система вызывает OB 80 (OB ошибок времени); если OB 80 не запрограммирован, то CPU переходит в состояние STOP.

Кроме контроля максимального времени цикла может быть гарантировано также минимальное время цикла сканирования. Операционная система задержит следующий запуск нового цикла (запись таблицы выходов образа процесса в модули вывода), пока не будет обеспечено минимальное время сканирования.

Диапазоны параметров задания максимального и минимального времени сканирования приведены в руководствах [/70/](#) и [/101/](#). Вы можете изменить настройку параметров с помощью STEP 7.

**Локальные данные OB1**

Следующая таблица описывает временные переменные (TEMP) для OB1.  
Имена переменных OB1 заданы по умолчанию.

Переменная	Тип	Описание
OB1_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: OB1 активен
OB1_SCAN_1	BYTE	В#16#01: завершение теплого рестарта В#16#02: завершение горячего рестарта В#16#03: завершение основного цикла В#16#04: завершение холодного рестарта
OB1_PRIORITY	BYTE	Класс приоритета 1
OB1_OB_NUMBR	BYTE	Номер OB (01)
OB1_RESERVED_1	BYTE	Резерв
OB1_RESERVED_2	BYTE	Резерв
OB1_PREV_CYCLE	INT	Длительность предыдущего цикла (мс)
OB1_MIN_CYCLE	INT	Минимальная длительность цикла (мс) со времени последнего запуска
OB1_MAX_CYCLE	INT	Максимальная длительность цикла (мс) со времени последнего запуска
OB1_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB



### 1.3 Организационные блоки прерываний по времени (OB10 – OB17)

#### Описание

STEP 7 предоставляет в распоряжение до восьми прерываний по времени (OB 10 - OB 17), которые могут запускаться однократно или периодически. Вы можете так параметризовать Ваше CPU при помощи SFC или STEP 7, что эти OB будут обрабатываться со следующими интервалами:

- Однократно
- Ежеминутно
- Ежечасно
- Ежедневно
- Еженедельно
- Ежемесячно
- В конце каждого месяца

#### Принцип действия OB прерываний по времени

Чтобы запустить прерывание по времени, его необходимо вначале установить, а потом активировать. Существует три следующих способа запуска:

- Автоматический запуск прерывания по времени. Он происходит как только Вы установили, а затем активировали прерывание по времени с помощью STEP 7. Следующая таблица показывает основные возможности активации прерывания по времени с помощью STEP 7.
- Вы устанавливаете прерывания по времени при помощи STEP 7, а затем активируете их вызовом SFC 30 "ACT\_TINT" в своей программе.
- Вы устанавливаете прерывание по времени вызовом SFC 28 "SET\_TINT", а затем активируете его вызовом SFC 30 "ACT\_TINT".

Интервал	Описание
Не активируется	Прерывание по времени не исполняется, даже когда оно загружено в CPU. Оно может быть активировано вызовом SFC30.
Активируется только один раз	OB прерывания по времени автоматически отменяется после того, как он исполняется однократно в указанное время. Ваша программа может с помощью SFC 28 заново установить прерывание по времени и вновь его активировать через SFC 30.
Активируется периодически	Когда происходит прерывание по времени, CPU рассчитывает момент его следующего старта исходя из текущего времени и периода.

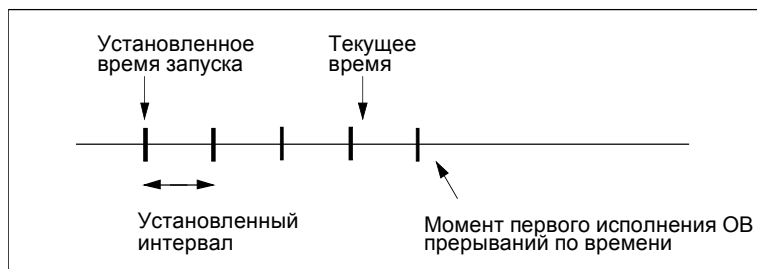
Поведение прерываний по времени при переводе часов вперед или назад описано в [/234/](#).

**Примечание**

Если Вы конфигурируете прерывание по времени на однократную обработку ОВ, то заданные значения даты и времени не могут находиться в прошлом (по отношению к часам реального времени CPU).

Если Вы конфигурируете прерывание по времени так, что соответствующий ОВ будет обрабатываться периодически, а дата и время лежат в прошлом, то прерывание по времени будет обработано в следующий надлежащий момент времени. Это иллюстрируется на следующем рисунке.

Прерывания по времени можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42. Более подробная информация содержится в главе 11.



**Условия, влияющие на ОВ прерываний по времени**

Т.к. прерывание по времени наступает через определенные интервалы времени, то определенные условия могут влиять на работу соответствующего ОВ во время исполнения Вашей программы. Следующая таблица представляет некоторые из таких условий и описывает их воздействие на обработку ОВ прерываний по времени.

Условие	Результат
Ваша программа вызывает SFC29 (CAN_TINT) и отменяет прерывание по времени.	Операционная система стирает стартовое событие (дату и время) прерывания по времени. Вам необходимо вновь установить стартовое событие и активировать его прежде, чем ОВ прерываний по времени вновь можно будет вызвать.
Ваша программа пыталась активировать ОВ прерывания по времени, который ко времени активации не был загружен в CPU.	Операционная система вызывает ОВ 85. Если ОВ 85 не был запрограммирован (загружен в CPU), то CPU переводится в состояние STOP.
При синхронизации или коррекции системных часов CPU Вы переставили время вперед и пропустили стартовое событие, т.е. дату или время для ОВ прерывания по времени.	Операционная система вызывает ОВ 80 и кодирует номер ОВ прерывания по времени и информацию стартового события в ОВ80. Операционная система обрабатывает затем ОВ прерывания по времени один раз независимо от того, сколько раз этот ОВ должен был бы обрабатываться. Информация о стартовом событии ОВ 80 показывает дату и время, когда ОВ прерывания по времени в первый раз был пропущен.
При синхронизации или коррекции системных часов CPU Вы перевели время назад так, что стартовое событие, т.е. дата или время для ОВ прерывания по времени повторяется.	Если ОВ прерывания по времени был активирован до того, как часы были переведены назад, то он не будет вновь вызываться.
CPU выполняет теплый или холодный рестарт.	Каждый ОВ прерывания по времени, который был сконфигурирован через SFC, вновь возвращается к конфигурации, заданной в STEP 7.
ОВ прерывания по времени еще обрабатывается, когда происходит стартовое событие для следующего интервала.	Операционная система вызывает ОВ 80. Если ОВ 80 не был запрограммирован, то CPU переходит в состояние STOP. Если ОВ80 загружен, то сначала исполняются ОВ80 и ОВ прерывания по времени, а затем исполняется второе затребованное прерывание по времени.

**Локальные данные ОВ прерываний по времени**

Следующая таблица описывает временные переменные (TEMP) одного из ОВ прерываний по времени. В качестве имен переменных выбраны имена по умолчанию ОВ 10.

Переменная	Тип	Описание
OB10_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11 = прерывание активно
OB10_STRT_INFO	BYTE	В#16#11: запрос на запуск для ОВ10 (В#16#12: запрос на запуск для ОВ11) : : (В#16#18: запрос на запуск для ОВ17)
OB10_PRIORITY	BYTE	Назначенный класс приоритета; по умолчанию 2
OB10_OB_NUMBR	BYTE	Номер ОВ (10 – 17)
OB10_RESERVED_1	BYTE	Резерв
OB10_RESERVED_2	BYTE	Резерв
OB10_PERIOD_EXE	WORD	ОВ обрабатывается с заданными интервалами: W#16#0000: однократно W#16#0201: ежеминутно W#16#0401: ежечасно W#16#1001: ежедневно W#16#1201: еженедельно W#16#1401: ежемесячно W#16#1801: ежегодно
OB10_RESERVED_3	INT	Резерв
OB10_RESERVED_4	INT	Резерв
OB10_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОВ.

## 1.4 Организационные блоки прерываний с задержкой (OB20 – OB23)

### Описание

S7 предоставляет в распоряжение до четырех OB (OB 20 – OB 23), которые исполняются после заданной задержки. Каждый OB прерывания с задержкой запускается посредством вызова SFC32 (SRT\_DINT). Время задержки является входным параметром SFC.

Когда Ваша программа вызывает функцию SFC32 (SRT\_DINT), то ей передается номер OB, время задержки и индивидуальный код пользователя. Через заданное время задержки стартует соответствующий OB. Имеется возможность отменить обработку прерывания с задержкой, которое еще не стартовало.

### Принцип действия OB прерываний с задержкой

По истечении времени задержки (его значение в миллисекундах передается блоку SFC32 вместе с номером OB) операционная система запускает соответствующий.

Чтобы использовать OB прерываний с задержкой, Вы должны выполнить следующие задачи:

- Вы должны вызвать SFC32 (SRT\_DINT).
- Вы должны загрузить OB прерываний с задержкой в CPU как часть своей программы.

OB прерываний с задержкой исполняются только в том случае, когда CPU находится в режиме RUN. Теплый или холодный рестарт стирает любые стартовые события для OB прерываний с задержкой. Если прерывание с задержкой еще не запущено, то для отмены его исполнения можно использовать SFC33 (CAN\_DINT).

Время задержки измеряется с точностью 1 мс. Время задержки может быть повторно запущено непосредственно после его окончания. С помощью SFC34 (QRY\_DINT) Вы можете опросить состояние прерывания с задержкой.

Операционная система вызывает OB асинхронных ошибок, если происходит одно из следующих событий:

- Если операционная система пытается запустить OB, который не загружен, а его номер Вы задали при вызове SFC32 "SRT\_DINT".
- Если следующее стартовое событие для запуска прерывания с задержкой наступает прежде, чем закончилась обработка текущего OB прерывания с задержкой.

Прерывания с задержкой можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

**Локальные данные ОВ прерываний с задержкой**

Следующая таблица описывает временные переменные (TEMP) ОВ прерывания с задержкой. В качестве имен переменных взяты имена по умолчанию ОВ20.

Переменная	Тип	Описание
OB20_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: прерывание активно
OB20_STRT_INF	BYTE	В#16#21: требование запуска для ОВ20 (В#16#22: требование запуска для ОВ21) (В#16#23: требование запуска для ОВ22) (В#16#24: требование запуска для ОВ23)
OB20_PRIORITY	BYTE	Назначенный класс приоритета: значения по умолчанию от 3 (ОВ20) до 6 (ОВ23)
OB20_OB_NUMBR	BYTE	Номер ОВ (20 – 23)
OB20_RESERVED_1	BYTE	Резерв
OB20_RESERVED_2	BYTE	Резерв
OB20_SIGN	WORD	Идентификатор пользователя: входной параметр SIGN из вызова SFC32 (SRT_DINT)
OB20_DTIME	TIME	Истекшая часть времени задержки в мс
OB20_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОВ

## 1.5 Организационные блоки циклических прерываний (ОВ30 – ОВ38)

### Описание

S7 представляет в распоряжение до девяти ОВ циклических прерываний (ОВ 30 – ОВ38), которые прерывают Вашу программу через фиксированные интервалы времени. Следующая таблица показывает установленные по умолчанию интервалы времени и классы приоритета для ОВ циклических прерываний.

Номер ОВ	Интервал по умолчанию	Класс приоритета по умолчанию
ОВ30	5 с	7
ОВ31	2 с	8
ОВ32	1 с	9
ОВ33	500 мс	10
ОВ34	200 мс	11
ОВ35	100 мс	12
ОВ36	50 мс	13
ОВ37	20 мс	14
ОВ38	10 мс	15

### Принцип действия ОВ циклических прерываний

Эквидистантные моменты запуска ОВ циклических прерываний определяются интервалом и фазовым сдвигом. Как связаны друг с другом момент запуска, периодичность и фазовый сдвиг, описано в [/234/](#).

#### Примечание

Вы должны позаботиться о том, чтобы время работы ОВ циклических прерываний было значительно меньше интервала времени, через который он вызывается. Если ОВ циклических прерываний еще не закончен, а в следствие истечения интервала времени должен обрабатываться вновь, запускается ОВ ошибок времени (ОВ 80). Циклическое прерывание, вызвавшее ошибку, будет отработано позднее.

Диапазон значений для периодичности, классов приоритета и фазового сдвига приводятся в технических данных CPU. Вы можете изменить настройки этих параметров с помощью STEP 7.

**Локальные данные ОБ циклических прерываний**

Следующая таблица временные (TEMP) переменные одного из ОБ циклических прерываний. В качестве имен переменных по умолчанию взяты имена ОБ35.

Переменная	Тип	Описание
OB35_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: прерывание активно
OB35_STRT_INF	BYTE	(В#16#31 : требование запуска для ОБ30) : В#16#36 : требование запуска для ОБ35 : (В#16#39 : требование запуска для ОБ38)
OB35_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию от 7 (ОБ30) до 15 (ОБ38)
OB35_OB_NUMBR	BYTE	Номер ОБ (30 – 38)
OB35_RESERVED_1	BYTE	Резерв
OB35_RESERVED_2	BYTE	Резерв
OB35_PHASE_OFFSET	WORD	Фазовый сдвиг [мс]
OB35_RESERVED_3	INT	Резерв
OB35_EXC_FREQ	INT	Интервал в миллисекундах
OB35_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОБ



## 1.6 Организационные блоки аппаратных прерываний (ОВ40 – ОВ47)

### Описание

S7 представляет в распоряжение до восьми независимых друг от друга аппаратных прерываний со своими собственными ОВ.

Назначая с помощью STEP 7 параметры, Вы для каждого сигнального модуля, который будет запускать аппаратные прерывания, указываете следующее:

- Какие каналы при каких условиях запускают аппаратное прерывание.
- Какой ОВ аппаратных прерываний ставится в соответствие отдельным группам каналов (по умолчанию все аппаратные прерывания обрабатываются ОВ40).

Для CP и FM эти параметры назначаются с помощью их собственного программного обеспечения.

Классы приоритета для отдельных ОВ аппаратных прерываний устанавливаются с помощью STEP 7.

### Принцип действия ОВ аппаратных прерываний

После запуска модулем аппаратного прерывания операционная система идентифицирует слот и соответствующий ОВ аппаратных прерываний. Если этот ОВ имеет более высокий приоритет, чем активный в данный момент класс приоритета, то он запускается. Соответствующее каналу квитирование выполняется после завершения этого ОВ аппаратных прерываний.

Если в промежутке времени между идентификацией и квитированием аппаратного прерывания на том же самом модуле возникает еще одно событие, которое вызывает аппаратное прерывание, то:

- Если событие наступает на том же самом канале, который до этого вызвал аппаратное прерывание, то новое прерывание теряется. Это иллюстрируется на следующем рисунке на примере канала модуля цифрового ввода. Запускающим событием является нарастающий фронт. ОВ аппаратных прерываний является ОВ40.



- Если событие происходит на другом канале того же самого модуля, то аппаратное прерывание в данный момент не может быть запущено. Однако это прерывание не теряется, а будет запущено после квитирования текущего активного прерывания (только для S7-400). В случае S7-300 аппаратное прерывание теряется, если вызывающее его событие закончилось раньше квитирования.

Если аппаратное прерывание запускается, а его ОВ в данный момент активен из-за аппаратного прерывания из другого модуля, то новый запрос регистрируется, а ОВ обрабатывается когда он будет свободен (только на S7-400). В случае S7-300 аппаратное прерывание теряется, если событие, вызывающее прерывание, не сохранилось после квитирования.

Аппаратные прерывания можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

Вы можете назначать параметры аппаратным прерываниям модуля не только при помощи STEP 7, но и при помощи SFC 55 – 57.

### Локальные данные ОВ аппаратных прерываний

Следующая таблица описывает временные (TEMP) переменные одного из ОВ аппаратных прерывания. В качестве имен переменных выбраны имена по умолчанию ОВ40.

Переменная	Тип	Описание
OB40_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#11: прерывание активно
OB40_STRT_INF	BYTE	V#16#41: прерывание через канал прерывания 1 V#16#42: прерывание через канал прерывания 2 (только для S7-400) V#16#43: прерывание через канал прерывания 3 (только для S7-400) V#16#44: прерывание через канал прерывания 4 (только для S7-400)
OB40_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию от 16 (OB40) до 23 (OB47)
OB40_OB_NUMBR	BYTE	Номер ОВ (40 – 47)
OB40_RESERVED_1	BYTE	Резерв
OB40_IO_FLAG	BYTE	Модуль ввода: V#16#54 Модуль вывода: V#16#55
OB40_MDL_ADDR	WORD	Логический базовый адрес модуля, который запустил прерывание
OB40_POINT_ADDR	DWORD	Для цифровых модулей: битовый массив с состояниями входов на модуле (бит 0 соответствует первому входу) Для аналоговых модулей (CP или FM): состояние прерывания модуля (для пользователя значения не имеет)
OB40_DATE_TIME	DATE_AND_TIME	Дата и время, когда ОВ был вызван

## 1.7 Организационный блок прерывания статуса (OB55)

### Примечание

OB прерывания статуса (OB55) доступен только для CPU с PDV1.

### Описание

Операционная система CPU вызывает OB55, если выполняется прерывание статуса от слота с ведомым устройством DPV1. Это происходит в случаях, когда компонент (модуль или стойка) из ведомых устройств DPV1 меняет свой рабочий режим, например, переходит от режима RUN к режиму STOP. Для получения подробной информации о событиях, вызывающих прерывание статуса, обратитесь к документации на DPV1 устройство.

### Локальные данные OB прерывания статуса

Следующая таблица описывает временные (TEMP) переменные OB прерывания статуса. В качестве имен переменных по умолчанию взяты имена OB55.

Переменная	Тип	Описание
OB55_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: (входное событие)
OB55_STRT_INF	BYTE	В#16#55: (требование запуска для OB55)
OB55_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию 2
OB55_OB_NUMBR	BYTE	Номер OB (55)
OB55_RESERVED_1	BYTE	Резерв
OB55_IO_FLAG	BYTE	Входной модуль: В#16#54 Выходной модуль: В#16#55
OB55_MDL_ADDR	WORD	Логический базовый адрес вызвавшего прерывание компонента (модуля)
OB55_LEN	BYTE	Размер блока данных
OB55_TYPE	BYTE	Идентификатор для типа "прерывание статуса"
OB55_SLOT	BYTE	Номер слота вызвавшего прерывание компонента (модуля)
OB55_SPEC	BYTE	Спецификатор <ul style="list-style-type: none"> <li>• Биты 0...1: спецификатор прерывания</li> <li>• Бит 2: Add_Ack (квитирование)</li> <li>• Биты 3...7: Seq.no. (порядковый номер)</li> </ul>
OB55_DATE_TIME	DATE_AND_TIME	Дата и время вызова OB

### Примечание

Вы можете получить полную дополнительную информацию по сообщениям прерывания, вызвав функцию SFB54 "RALRM" с OB55.

## 1.8 Организационный блок прерывания модификации (OB56)

### Примечание

OB прерывания модификации (OB56) доступен только для CPU с PDV1.

### Описание

Операционная система CPU вызывает OB56, если выполняется прерывание модификации от слота с ведомым устройством DPV1. Это происходит в случаях, когда меняются параметры для слота ведомого устройства DPV1 (посредством локального или удаленного доступа). Для получения подробной информации о событиях, вызывающих прерывание статуса, обратитесь к документации на DPV1 устройство.

### Локальные данные OB прерывания модификации (update)

Следующая таблица описывает временные (TEMP) переменные OB прерывания модификации. В качестве имен переменных по умолчанию взяты имена OB56.

Переменная	Тип	Описание
OB56_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#11: (входное событие)
OB56_STRT_INF	BYTE	V#16#56: (требование запуска для OB56)
OB56_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию 2
OB56_OB_NUMBR	BYTE	Номер OB (56)
OB56_RESERVED_1	BYTE	Резерв
OB56_IO_FLAG	BYTE	Входной модуль: V#16#54 Выходной модуль: V#16#55
OB56_MDL_ADDR	WORD	Логический базовый адрес вызвавшего прерывание компонента (модуля)
OB56_LEN	BYTE	Длина блока данных
OB56_TYPE	BYTE	Идентификатор для типа "прерывание модифик."
OB56_SLOT	BYTE	Номер слота вызвавшего прерывание компонента (модуля)
OB56_SPEC	BYTE	Спецификатор <ul style="list-style-type: none"> <li>• Биты 0...1: спецификатор прерывания</li> <li>• Бит 2: Add_Ack (квитирование)</li> <li>• Биты 3...7: Seq.no. (порядковый номер)</li> </ul>
OB56_DATE_TIME	DATE_AND_TIME	Дата и время вызова OB

### Примечание

Вы можете получить полную дополнительную информацию по сообщениям прерывания, вызвав функцию SFB54 "RALRM" с OB56.

## 1.9 Организационный блок прерывания производителя (OB57)

---

### Примечание

OB прерывания статуса (OB57) доступен только для CPU с PDV1.

---

### Описание

Операционная система CPU вызывает OB57, если выполняется прерывание, определенное производителем устройства, от слота с ведомым устройством DPV1.

### Локальные данные OB прерывания, определенного производителем

Следующая таблица описывает временные (TEMP) переменные OB прерывания, определенного производителем. В качестве имен переменных по умолчанию взяты имена OB57.

Переменная	Тип	Описание
OB57_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: (входное событие)
OB57_STRT_INF	BYTE	В#16#57: (требование запуска для OB57)
OB57_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию 2
OB57_OB_NUMBR	BYTE	Номер OB (57)
OB57_RESERVED_1	BYTE	Резерв
OB57_IO_FLAG	BYTE	Входной модуль: В#16#54 Выходной модуль: В#16#55
OB57_MDL_ADDR	WORD	Логический базовый адрес вызвавшего прерывание компонента (модуля)
OB57_LEN	BYTE	Длина блока данных
OB57_TYPE	BYTE	Идентификатор для типа "прерывание, определенное производителем"
OB57_SLOT	BYTE	Номер слота вызвавшего прерывание компонента (модуля)
OB57_SPEC	BYTE	Спецификатор <ul style="list-style-type: none"> <li>• Биты 0...1: спецификатор прерывания</li> <li>• Бит 2: Add_Ack (квитирование)</li> <li>• Биты 3...7: Seq.no. (порядковый номер)</li> </ul>
OB57_DATE_TIME	DATE_AND_TIME	Дата и время вызова OB

---

### Примечание

Вы можете получить полную дополнительную информацию по сообщениям прерывания, вызвав функцию SFB54 "RALRM" с OB57.

---

## 1.10 Организационный блок мультипроцессорных прерываний (OB60)

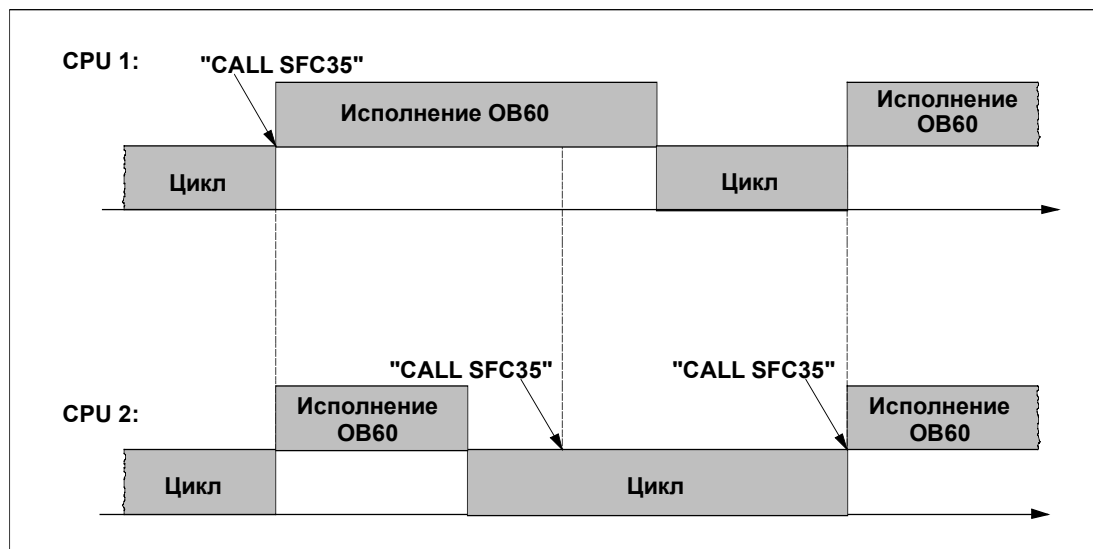
### Описание

С помощью мультипроцессорного прерывания Вы можете обеспечить синхронную реакцию соответствующих CPU на событие при мультипроцессорной обработке. В отличие от аппаратных прерываний, запускаемых сигнальными модулями, мультипроцессорное прерывание может выдаваться только центральными процессорами.

### Принцип действия OB мультипроцессорных прерываний

Мультипроцессорное прерывание инициируется вызовом SFC 35 "MP\_ALM". При мультипроцессорной обработке это приводит к синхронизированному старту OB60 на всех установленных CPU данного шинного сегмента, если Вы не заблокировали OB60 (с помощью SFC 39 "DIS\_IRT") или не задержали его (с помощью SFC 41 "DIS\_AIRT"). Если Вы не загрузили OB 60 в CPU, то соответствующий CPU немедленно возвращается в последний класс приоритета перед прерыванием и продолжает там обработку программы. В однопроцессорном режиме и при использовании сегментированных стоек OB60 запускается только на том CPU, на котором Вы вызвали SFC35 "MP\_ALM".

Когда Ваша программа вызывает SFC 35 "MP\_ALM", Вы передаете ей идентификатор задания. Этот идентификатор передается на все. Это позволяет Вам реагировать на конкретное событие. Если OB 60 на отдельных CPU запрограммирован по-разному, то время его обработки может оказаться различным. В этом случае CPU возвращаются к прерванному классу приоритета в различные моменты времени. Если следующее мультипроцессорное прерывание выдается одним CPU в то время, как другой еще занят обработкой OB 60 предыдущего мультипроцессорного прерывания, то запуска OB 60 не происходит ни на запрашивающем, ни на каком другом CPU, принадлежащем данному шинному сегменту. Это иллюстрируется на следующем рисунке на примере двух CPU. Вы извещаетесь о результате с помощью значения вызванной функции SFC35.



### Локальные данные ОБ мультипроцессорных прерываний

Следующая таблица описывает временные (TEMP) переменные ОБ мультипроцессорных прерываний. В качестве имен переменных взяты имена по умолчанию OB60

Переменная	Тип	Описание
OB60_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#11: прерывание активно
OB60_STRT_INF	BYTE	V#16#61: Мультипроцессорное прерывание запущено собственным CPU V#16#62: Мультипроцессорное прерывание запущено другим CPU
OB60_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию 25
OB60_OB_NUMBR	BYTE	Номер ОБ: 60
OB60_RESERVED_1	BYTE	Резерв
OB60_RESERVED_2	BYTE	Резерв
OB60_JOB	INT	Идентификатор задания: входная переменная JOB SFC35 "MP_ALM"
OB60_RESERVED_3	INT	Резерв
OB60_RESERVED_4	INT	Резерв
OB60_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОБ.

## 1.11 Организационный блок синхронного циклического прерывания (OB61)

### Описание

Синхронное циклическое прерывание дает Вам возможность обеспечить синхронный запуск программ в цикле DP-периферии. OB61 выполняет роль интерфейсного OB для синхронного циклического прерывания TSAL1. Вы можете устанавливать приоритетный класс для OB1 с номерами 0 (OB не выбран) и от 2 до 26.

### Локальные данные OB синхронного циклического прерывания

Следующая таблица описывает временные (TEMP) переменные OB синхронного циклического прерывания. В качестве имен переменных взяты имена по умолчанию OB61.

Переменная	Тип	Описание
OB61_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#11: прерывание активно
OB61_STRT_INF	BYTE	В#16#64: требование запуска для OB61
OB61_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию 25
OB61_OB_NUMBR	BYTE	Номер OB: 61
OB61_RESERVED_1	BYTE	Резерв
OB61_RESERVED_2	BYTE	Резерв
OB61_GC_VIOL	BOOL	GC-нарушение
OB61_FIRST	BOOL	Первый вызов после запуска или после режима STOP
OB61_MISSED_EXEC	BYTE	Число невыполненных запусков OB61 после последнего выполнения OB61
OB61_DP_ID	BYTE	Идентификатор (ID) ведущего DP-устройства
OB61_RESERVED_3	BYTE	Резерв
OB61_RESERVED_4	BYTE	Резерв
OB61_RESERVED_5	WORD	Резерв
OB61_DATE_TIME	DATE_AND_TIME	Дата и время вызова OB



## 1.12 ОВ ошибок резервирования входов/выходов (ОВ70)

---

### Примечание

ОВ ошибок резервирования входов/выходов (ОВ70) может использоваться только в H CPU.

---

### Описание

Операционная система H CPU вызывает ОВ70, когда имеет место потеря резервирования в PROFIBUS DP (например, когда происходит отказ шины активного ведущие DP-устройства или ошибка в интерфейсном модуле ведомые DP-устройства) или когда переключается активное ведущее устройство DP ведомых DP-устройств с подключенными входами/выходами.

Если ОВ70 не запрограммирован и происходит стартовое событие, то CPU не переходит в состояние STOP. Если ОВ70 загружен и H-система находится в резервном режиме, то ОВ70 выполняется в обоих CPU. H-система остается в резервном режиме.

### Локальные данные ОВ ошибок резервирования входов/выходов

Следующая таблица содержит временные (TEMP) переменные ОВ ошибок резервирования входов/выходов. Выбранные имена переменных являются заданными по умолчанию именами ОВ70.

Переменная	Тип	Характеристика
OB70_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#78: наступающее событие V#16#79: уходящее событие
OB70_FLT_ID	BYTE	Код ошибки (возможные значения): V#16#A2, V#16#A3
OB70_PRIORITY	BYTE	Назначенный класс приоритета: по умолчанию 25
OB70_OB_NUMBR	BYTE	Номер ОВ (70)
OB70_INFO_1	WORD	Зависит от кода ошибки
OB70_INFO_2	WORD	Зависит от кода ошибки
OB70_INFO_3	WORD	Зависит от кода ошибки
OB70_RESERVED_1	WORD	Резерв
OB80_DATE_TIME	DATE_AND_TIME	Дата и время, когда вызывался ОВ

Следующая таблица показывает, какое событие привело к запуску OB70.

OB70_FLT_ID	Стартовое событие OB70
V#16#A0	Переключился резервный интерфейсный модуль DP вследствие отказа активного ведомые DP-устройства.
V#16#A1	Переключился резервный интерфейсный модуль DP вследствие отказа активного ведущие DP-устройства.
V#16#A2	Отказ пассивного ведущие DP-устройства.
V#16#A3	Отказ пассивного ведомые DP-устройства.

Переменные, зависящие от кода ошибки, имеют следующее значение:

- Коды ошибки V#16#A2
 

OB70_INFO_1:		Логический базовый адрес неисправного ведущие DP-устройства
OB70_INFO_2:		Резерв
OB70_INFO_3:	Биты с 0 по 7:	Резерв
	Биты с 8 по 15:	Идентификатор master-системы DP неисправного ведущие DP-устройства
- Коды ошибки V#16#A3
 

OB70_INFO_1:		Логический базовый адрес ведущие DP-устройства
OB70_INFO_2:		Неисправное ведомое DP-устройство:
	Биты с 0 по 14:	Логический базовый адрес, если используется slave-устройство S7, либо диагностический адрес, если используется стандартное ведомое DP-устройство.
	Бит 15:	Идентификация входов/выходов
		Неисправное ведомое DP-устройство:
OB70_INFO_3:	Биты с 0 по 7:	Номер станции DP
	Биты с 8 по 15:	Идентификатор master-системы DP

## 1.13 ОВ ошибок резервирования CPU (OB72)

### Примечание

ОВ ошибок резервирования CPU (OB72) существует только у H CPU.

### Описание

Операционная система H CPU вызывает OB72, когда происходит одно из следующих событий:

- Потеря резервирования CPU
- Переключение на резервное ведущее устройство
- Ошибка синхронизации
- Ошибка в модуле синхронизации
- Прерывание обновления
- Ошибка сравнения (например, RAM, PIQ)

OB72 выполняется всеми CPU, которые находятся в режиме RUN или STARTUP, после соответствующего стартового события.

### Локальные данные ОВ ошибок резервирования CPU

Следующая таблица содержит временные (TEMP) переменные ОВ ошибок резервирования CPU. Заданные по умолчанию имена OB72 были использованы как имена переменных.

Переменная	Тип	Характеристика
OB72_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#73, В#16#75, В#16#79, В#16#78
OB72_FLT_ID	BYTE	Код ошибки (возможные значения: В#16#01, В#16#02, В#16#03, В#16#20, В#16#21, В#16#22, В#16#23, В#16#31, В#16#33, В#16#34, В#16#35, В#16#40, В#16#41, В#16#42, В#16#43, В#16#44, В#16#50, В#16#51, В#16#52, В#16#53, В#16#54, В#16#55, В#16#56, В#16#C1)
OB72_PRIORITY	BYTE	Назначенный класс приоритета: значение по умолчанию 28
OB72_OB_NUMBR	BYTE	Номер ОВ (72)
OB72_INFO_1	WORD	(Не имеет значения для пользователя)
OB72_INFO_2	WORD	(Не имеет значения для пользователя)
OB72_INFO_3	WORD	(Не имеет значения для пользователя)
OB72_RESERVED_1	WORD	Резерв
OB82_DATE_TIME	DATE_AND_TIME	Дата и время, когда вызывался ОВ

Следующая таблица показывает, какое событие привело к запуску OB72.

OB72_FLT_ID	Стартовое событие OB72
V#16#01	Потеря резервирования (1 из 2) вследствие запуска CPU
V#16#02	Потеря резервирования (1 из 2) вследствие перехода в STOP в резерве, запущенном пользователем
V#16#03	H-система (1 из 2) переключилась в резервный режим
V#16#20	Ошибка при сравнении RAM
V#16#21	Ошибка при сравнении выходного значения образа процесса
V#16#22	Ошибка при сравнении битов памяти, таймеров или счетчиков
V#16#31	Переключение на резервное ведущее устройство вследствие отказа master-устройства
V#16#33	Переключение на резервное ведущее устройство вследствие вмешательства оператора
V#16#34	Переключение на резервное ведущее устройство вследствие проблемы соединения
V#16#35	Переключение на резервное ведущее устройство, запущенное SFC90 "H_CTRL".
V#16#40	Ошибка синхронизации в программе пользователя из-за истекшего времени ожидания
V#16#41	Ошибка синхронизации в программе пользователя из-за ожидания в различных точках синхронизации
V#16#42	Ошибка синхронизации в операционной системе из-за ожидания в различных точках синхронизации
V#16#43	Ошибка синхронизации в операционной системе из-за истекшего времени ожидания
V#16#50	Нет модуля синхронизации (SYNC)
V#16#51	Изменение в модуле синхронизации без включения питания
V#16#52	Модуль синхронизации удален/вставлен
V#16#53	Изменение в модуле синхронизации без сброса
V#16#54	Модуль синхронизации: номер стойки назначен дважды
V#16#55	Ошибка модуля синхронизации/устранена
V#16#56	В модуле синхронизации установлен запрещенный номер стойки
V#16#C1	Прерывание обновления

## 1.14 ОВ ошибок резервирования соединений (ОВ73)

### Примечание

ОВ ошибок резервирования соединений (ОВ73) существует только в встроенных программах версии 2.0.x для CPU417-4H.

### Описание

Операционная система H CPU вызывает ОВ73, если происходит потеря резервирования в отказоустойчивой S7-системе связи. Подробную информацию по отказоустойчивым S7-системам связи в S7 см. в "S7-400 H Programmable Controller, Fault-Tolerant Systems." ["Программируемые контроллеры S7-400 H. Отказоустойчивые системы"]. Если вновь происходит потеря резервирования (в других отказоустойчивых S7-каналах связи), то новые блоки ОВ73 не запускаются.

Запуск других ОВ73 не будет происходить до тех пор, пока не будет восстановлено резервирование для всех S7 соединений, в которых возникла неисправность.

CPU не перейдет в режим STOP, если происходит стартовое событие, а ОВ73 не запрограммирован.

### Локальные данные ОВ ошибок резервирования соединения (канала связи)

Следующая таблица содержит временные (TEMP) переменные ОВ ошибок резервирования канала связи. Заданные по умолчанию имена ОВ73 были использованы как имена переменных.

Переменная	Тип	Характеристика
ОВ73_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#73, В#16#72
ОВ73_FLT_ID	BYTE	Код ошибки (возможные значения: В#16#E0)
ОВ73_PRIORITY	BYTE	Назначенный класс приоритета: значение по умолчанию 25
ОВ73_OB_NUMBR	BYTE	Номер ОВ (73)
ОВ73_RESERVED_1	WORD	Резерв
ОВ73_INFO_1	WORD	(Не имеет значения для пользователя)
ОВ73_INFO_2	WORD	(Не имеет значения для пользователя)
ОВ73_INFO_3	WORD	(Не имеет значения для пользователя)
ОВ73_DATE_TIME	DATE_AND_TIME	Дата и время вызова ОВ

Следующая таблица показывает, какое событие привело к запуску OB73.

<b>OB73_FLT_ID</b>	<b>Стартовое событие OB73</b>
B#16#E0	Потеря резервирования в удаленном (й) соединении / задаче

## 1.15 Организационный блок ошибок времени (OB80)

### Описание

Операционная система CPU S7-300 вызывает OB80, когда при обработке какого-либо OB возникает одна из следующих ошибок: превышение времени цикла, ошибка квитирования при исполнении OB, перевод часов вперед, так что пропускается время запуска OB. Если, например, стартовое событие для OB циклических прерываний возникает до того, как была закончена обработка предыдущего вызова, то операционная система вызывает OB80. Если OB 80 не был запрограммирован, то CPU переходит в состояние STOP. OB ошибок времени можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Примечание

Если OB 80 в одном и том же цикле вызывается дважды из-за превышения времени цикла, то CPU переходит в состояние STOP. Вы можете этому воспрепятствовать вызовом SFC43 "RE\_TRIGR" в подходящей точке программы.

### Локальные данные OB ошибок времени

Следующая таблица описывает временные (TEMP) переменные OB ошибок времени. В качестве имен переменных взяты имена по умолчанию OB80.

Переменная	Тип	Описание
OB80_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#35
OB80_FLT_ID	BYTE	Код ошибки (возможные значения В#16#01, В#16#02, В#16#05, В#16#06, В#16#07)
OB80_PRIORITY	BYTE	Класс приоритета: 26 (режим RUN) или 28 (режим STARTUP) На резервном CPU Н-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2: номер стойки Бит 3: 0=резервный CPU, 1=основной CPU Биты 4 – 7: 1111
OB80_OB_NUMBR	BYTE	Номер OB (80)
OB80_RESERVED_1	BYTE	Резерв
OB80_RESERVED_2	BYTE	Резерв
OB80_ERROR_INFO	WORD	Информация об ошибке: зависит от кода ошибки
OB80_ERR_EV_CLASS	BYTE	Класс события, которое вызвало ошибку
OB80_ERR_EV_NUM	BYTE	Номер события, которое вызвало ошибку
OB80_OB_PRIORITY	BYTE	Класс приоритета OB, который был активен, когда произошла ошибка
OB80_OB_NUM	BYTE	Номер OB, который был активен, когда произошла ошибка
OB80_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

Переменные, зависящие от кода ошибки, имеют следующие значения:

• Код ошибки V#16#01		Превышено время цикла.
OB80_ERROR_INFO:		Длительность последнего цикла (мс).
OB80_ERR_EV_CLASS:		Класс события, которое вызвало прерывание.
OB80_ERR_EV_NUM:		Номер события, которое вызвало прерывание.
OB80_OB_PRIORITY:		Класс приоритета, активный на момент прерывания.
• Код ошибки	V#16#02	Затребованный ОБ еще выполняется.
	V#16#07	Переполнение буфера вызовов ОБ для текущего класса приоритета (каждый запрос старта ОБ для класса приоритета заносится в соответствующий буфер вызовов ОБ; по завершении ОБ эта запись удаляется. Если же для некоторого класса приоритета количество вызовов ОБ превысило максимально допустимое число записей соответствующего буфера вызовов, то вызывается ОБ 80 с кодом ошибки V#16#07).
OB80_ERROR_INFO:		Соответствующая временная переменная запрошенного блока, определяемая с помощью OB80_ERR_EV_CLASS и OB80_ERR_EV_NUM.
OB80_ERR_EV_CLASS:		Класс события, вызвавшего прерывание.
OB80_ERR_EV_NUM:		Номер события, вызвавшего прерывание.
OB80_OB_PRIORITY:		Класс приоритета, активный на момент прерывания.
OB80_OB_NUM:		ОБ, активный на момент прерывания.
• Код ошибки	V#16#05	Истекшее прерывание по времени из-за перевода часов вперед.
	V#16#06	Истекшее прерывание по времени после возвращения в RUN после HOLD.
OB80_ERROR_INFO	Бит 0 установлен	Для прерывания по времени №0 стартовая точка лежит в прошлом.
	:	
	:	
	Бит 7 установлен:	Для прерывания по времени №7 стартовая точка лежит в прошлом.
	Биты 8 – 15:	не используются
OB80_ERR_EV_CLASS:		не используется
OB80_ERR_EV_NUM:		не используется
OB80_OB_PRIORITY:		не используется
OB80_OB_NUM:		не используется



## 1.16 Организационный блок неисправностей источника питания (OB81)

### Описание

Операционная система CPU S7-300 вызывает OB81, когда происходит событие, вызванное ошибкой или сбоем, связанным с источником питания (только для S7-400) или буферной батареей (при наступающем и при уходящем событии).

В отличие от OB для других асинхронных ошибок CPU в данном случае не переходит в режим STOP, если OB 81 не был запрограммирован.

OB неисправностей источника питания можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Локальные данные OB неисправностей источника питания

Следующая таблица описывает временные (TEMP) переменные OB неисправностей источника питания. В качестве имен переменных взяты имена по умолчанию OB81.

Переменная	Тип	Описание
OB81_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#38: наступающее событие V#16#39: уходящее событие
OB81_FLT_ID	BYTE	Код ошибки: (возможные значения V#16#21, V#16#22, V#16#23, V#16#25, V#16#26, V#16#27, V#16#31, V#16#32, V#16#33)
OB81_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для режима RUN) или 28 (режим STARTUP) На резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2:       Номер стойки Бит 3:            0=резервный CPU, 1=основной CPU Биты 4 – 7:       1111
OB81_OB_NUMBR	BYTE	Номер OB (81)
OB81_RESERVED_1	BYTE	Резерв
OB81_RESERVED_2	BYTE	Резерв
OB81_MDL_ADDR	INT	Резерв
OB81_RESERVED_3	BYTE	Имеет значение только для кодов ошибок V#16#31, V#16#32 и V#16#33
OB81_RESERVED_4	BYTE	
OB81_RESERVED_5	BYTE	
OB81_RESERVED_6	BYTE	
OB81_DATE_TIME	DATE_AND_TIME	Дата и время, когда OB был вызван.

Переменные OB81\_RESERVED\_i,  $3 \leq i \leq 6$  указывают на стойки расширения, на которых возникла неисправность или произошло восстановление буферной батареи (код ошибки V#16#31), резервного напряжения (код ошибки V#16#32) или источника питания 24 В (код ошибки V#16#33). Следующая таблица показывает, какой бит какой стойке расширения соответствует в переменных OB81\_RESERVED\_i,  $3 \leq i \leq 6$ .

	<b>OB81_RESERVED_6</b>	<b>OB81_RESERVED_5</b>	<b>OB81_RESERVED_4</b>	<b>OB81_RESERVED_3</b>
Бит 0	Резерв	8-я стойка расширения	16-я стойка расширения	Резерв
Бит 1	1-я стойка расширения	9-я стойка расширения	17-я стойка расширения	Резерв
Бит 2	2-я стойка расширения	10-я стойка расширения	18-я стойка расширения	Резерв
Бит 3	3-я стойка расширения	11-я стойка расширения	19-я стойка расширения	Резерв
Бит 4	4-я стойка расширения	12-я стойка расширения	20-я стойка расширения	Резерв
Бит 5	5-я стойка расширения	13-я стойка расширения	21-я стойка расширения	Резерв
Бит 6	6-я стойка расширения	14-я стойка расширения	Резерв	Резерв
Бит 7	7-я стойка расширения	15-я стойка расширения	Резерв	Резерв

Биты в переменной OB81\_RESERVED\_i имеют следующее значение (для соответствующих стоек расширения):

При наступлении события помечаются стойки расширения (устанавливаются соответствующие биты), у которых вышли из строя хотя бы одна буферная батарея, или резервное питание, или источник питания 24 В. Стойки расширения, у которых до этого хотя бы одна батарея, или резервное питание, или источник питания 24 В уже вышли из строя, более не индицируются.

Когда событие устраняется и резервирование восстанавливается по крайней мере на одной стойке расширения, то об этом сообщается (устанавливаются соответствующие биты).

Переменная OB81\_FLT\_ID имеет следующее значение:

- V#16#21: Хотя бы одна буферная батарея центральной стойки разрядилась/ проблема устранена (BATTF)
- V#16#22: Резервное напряжение в центральной стойке вышло из строя/ проблема устранена (BAF)
- V#16#23: Неисправность источника питания 24 В в центральной стойке / проблема устранена.
- V#16#25: Хотя бы одна буферная батарея по крайней мере в одной резервной центральной стойке разрядилась / проблема устранена (BATTF)
- V#16#26: Резервное напряжение по крайней мере в одной резервной центральной стойке вышло из строя/ проблема устранена (BAF)
- V#16#27: Неисправность источника питания 24 В по крайней мере в одной резервной центральной стойке
- V#16#31: Хотя бы одна буферная батарея по крайней мере в одной стойке расширения разрядилась/ проблема устранена (BATTF).
- V#16#32: Резервное напряжение по крайней мере в одной стойке расширения вышло из строя/ проблема устранена (BAF)
- V#16#33: Неисправность источника питания 24 В по крайней мере в одной стойке расширения / проблема устранена.

## 1.17 Организационный блок диагностических прерываний (OB82)

### Описание

Если модуль, обладающий диагностическими свойствами, которому Вы разрешили диагностические прерывания, распознает ошибку, он выдает на CPU запрос на диагностическое прерывание (как при наступающем, так и при уходящем событии). Затем операционная система вызывает OB82.

OB 82 содержит в своих локальных переменных как логический базовый адрес, так и четырехбайтовую диагностическую информацию неисправного модуля (см. следующую таблицу).

Если OB 82 не был запрограммирован, то CPU переходит в состояние STOP.

OB диагностических прерываний можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Локальные данные OB диагностических прерываний

Следующая таблица описывает временные (TEMP) переменные OB диагностических прерываний. В качестве имен переменных взяты имена по умолчанию OB82.

Переменная	Тип	Описание
OB82_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#38: уходящее событие V#16#39: наступающее событие
OB82_FLT_ID	BYTE	Код ошибки (V#16#42)
OB82_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для режима RUN) или 28 (режим STARTUP) На резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2: Номер стойки Бит 3: 0=резервный CPU, 1=основной CPU Биты 4 – 7: 1111
OB82_OB_NUMBR	BYTE	Номер OB (82)
OB82_RESERVED_1	BYTE	Резерв
OB82_IO_FLAG	BYTE	Модуль ввода: V#16#54 Модуль вывода: V#16#55
OB82_MDL_ADDR	INT	Базовый логический адрес модуля, в котором произошел отказ.
OB82_MDL_DEFECT	BOOL	Модуль неисправен
OB82_INT_FAULT	BOOL	Внутренняя ошибка
OB82_EXT_FAULT	BOOL	Внешняя ошибка
OB82_PNT_INFO	BOOL	Ошибка канала
OB82_EXT_VOLTAGE	BOOL	Внешнее питание вышло из строя
OB82_FLD_CONNCTR	BOOL	Не вставлен фронт-штекер
OB82_NO_CONFIG	BOOL	Модуль не сконфигурирован
OB82_CONFIG_ERR	BOOL	Неверные параметры в модуле

Переменная	Тип	Описание
OB82_MDL_TYPE	BYTE	Биты 0 – 3: Класс модуля Бит 4: Имеется информация канала Бит 5: Имеется информация пользователя Бит 6: Диагностическое прерывание из-за замены Бит 7: Резерв
OB82_SUB_MDL_ERR	BOOL	Субмодуль отсутствует или неисправен
OB82_COMM_FAULT	BOOL	Неисправность связи
OB82_MDL_STOP	BOOL	Рабочий режим (0: RUN, 1: STOP)
OB82_WTCH_DOG_FLT	BOOL	Сработал контроль времени
OB82_INT_PS_FLT	BOOL	Отказ внутреннего источника питания
OB82_PRIM_BATT_FLT	BOOL	Разрядилась батарея
OB82_BCKUP_BATT_FLT	BOOL	Выход из строя всего резервирования
OB82_RESERVED_2	BOOL	Резерв
OB82_RACK_FLT	BOOL	Выход из строя стойки расширения
OB82_PROC_FLT	BOOL	Выход из строя процессора
OB82_EPROM_FLT	BOOL	Сбой СППЗУ
OB82_RAM_FLT	BOOL	Сбой ОЗУ
OB82_ADU_FLT	BOOL	Ошибка АЦП/ЦАП
OB82_FUSE_FLT	BOOL	Выход из строя плавкого предохранителя
OB82_HW_INTR_FLT	BOOL	Потеряно аппаратное прерывание
OB82_RESERVED_3	BOOL	Резерв
OB82_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

## 1.18 Организационный блок снятия/установки модулей (OB83)

### Описание

Установка и снятие модулей контролируется внутри системы каждую секунду. Чтобы установка или снятие модуля распознавались CPU, необходимо чтобы между установкой и снятием прошло как минимум две секунды.

Каждая установка или снятие сконфигурированного модуля в режимах RUN, STOP и STARTUP (не разрешено удаление в этих режимах для блоков питания, CPU, адаптерных модулей и IM) приводит к прерыванию снятия/установки. Это прерывание вызывает у соответствующего CPU запись в диагностический буфер и в список состояний системы. Кроме того, в режиме RUN осуществляется запуск OB снятия/установки. Если этот OB не был запрограммирован, то CPU переходит в состояние STOP.

OB снятия/установки можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Принцип действия OB83

При снятии сконфигурированного модуля в режиме RUN запускается OB83. Так как существование модулей контролируется только с интервалом в одну секунду, то сначала может быть обнаружена ошибка доступа при прямом обращении к модулю или обновлении образа процесса.

При установке модуля в сконфигурированный слот в режиме RUN операционная система проверяет соответствие вставленного модуля зарегистрированной конфигурации. Затем запускается OB 83 и при совпадении типов модулей осуществляется параметризация.

**Локальные данные OB83**

Следующая таблица описывает временные (TEMP) переменные OB снятия/установки. В качестве имен переменных взяты имена по умолчанию OB83.

Переменная	Тип данных	Описание
OB83_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#38: модуль установлен В#16#39: модуль снят или не реагирует
OB83_FLT_ID	BYTE	Код ошибки (возможные значения В#16#61, В#16#63 или В#16#64)
OB83_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для режима RUN) или 28 (режим STARTUP) На резервном CPU Н-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2: Номер стойки Бит 3: 0=резервный CPU, 1=основной CPU Биты 4 – 7: 1111
OB83_OB_NUMBR	BYTE	Номер OB (83)
OB83_RESERVED_1	BYTE	Резерв
OB83_MDL_TD	BYTE	Область: В#16#54: периферийные входы (PI) В#16#55: периферийные выходы (PQ)
OB83_MDL_ADDR	WORD	Логический базовый адрес соответствующего модуля
OB83_RACK_NUM	WORD	Номер стойки или номер станции DP (младший байт) и идентификатор ведущей DP-системы (старший байт)
OB83_MDL_TYPE	WORD	Тип соответствующего модуля (W#16#0005: аналоговый модуль, W#16#0008: функциональный модуль, W#16#000C: CP, W#16#000F: цифровой модуль)
OB83_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

Переменная OB83\_MDL\_TYPE имеет следующее значение в зависимости от кода ошибки:

- Код ошибки В#16#61  
Модуль установлен, тип модуля верен (для класса событий В#16#38)  
Модуль снят или не реагирует (для класса событий В#16#39)  
OB83\_MDL\_TYPE: Фактический тип модуля
- Код ошибки В#16#63:  
OB83\_MDL\_TYPE: Фактический тип модуля
- Код ошибки В#16#64:  
Модуль установлен, однако неисправен (не читается идентификатор типа)  
OB83\_MDL\_TYPE: Заданный в конфигурации тип модуля
- Код ошибки В#16#65:  
Модуль установлен, однако, имеется ошибка в назначении параметров модуля  
OB83\_MDL\_TYPE: Фактический тип модуля

## 1.19 Организационный блок аппаратных ошибок CPU (OB84)

### Описание

Операционная система CPU вызывает OB 84, если распознается ошибка в интерфейсе с сетью MPI, с внутренней коммуникационной шиной (К-шиной) или в сопряжении с интерфейсным модулем для децентрализованной периферии.

Если возникла такая ошибка, а OB 84 не был запрограммирован, то CPU переходит в состояние STOP.

OB аппаратных ошибок CPU можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Локальные данные OB аппаратных ошибок

Следующая таблица содержит временные (TEMP) переменные OB аппаратных ошибок CPU. В качестве имен переменных взяты имена по умолчанию OB84.

Переменная	Тип	Описание
OB84_EV_CLASS	BYTE	Класс события и идентификатор: B#16#38: уходящее событие B#16#39: наступающее событие
OB84_FLT_ID	BYTE	Код ошибок (B#16#81)
OB84_PRIORITY	BYTE	Класс приоритета: 26 (по умолчанию для режима RUN или 28 (режим STARTUP) На резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2: Номер стойки Бит 3: 0=резервный CPU, 1=основной CPU Биты 4 – 7: 1111
OB84_OB_NUMBR	BYTE	Номер OB (84)
OB84_RESERVED_1	BYTE	Резерв
OB84_RESERVED_2	BYTE	Резерв
OB84_RESERVED_3	WORD	Резерв
OB84_RESERVED_4	DWORD	Резерв
OB84_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

## 1.20 Организационный блок ошибок класса приоритета (OB85)

### Описание

Операционная система CPU вызывает OB85 всякий раз, когда происходит одно из следующих событий:

- Стартовое событие для OB, который не был загружен.
- Ошибка при обращении операционной системы к модулю.
- Ошибка доступа к входам/выходам во время обновления образа процесса системой.

### Примечание

Если OB85 не был запрограммирован, то в случае обнаружения одной из этих ошибок CPU переходит в состояние STOP.

Вы можете заблокировать или задержать и вновь разблокировать OB ошибок класса приоритета, используя SFC с номерами от 39 до 42.

### Локальные данные для OB ошибок класса приоритета

Следующая таблица описывает временные (TEMP) переменные для OB ошибок класса приоритета. Имена переменных являются заданными по умолчанию именами OB85.

Переменная	Тип	Характеристика
OB85_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#35 V#16#38 (только с кодами ошибок V#16#B3 и V#16#B4) V#16#39 (только с кодами ошибок V#16#B1 и V#16#B2)
OB85_FLT_ID	BYTE	Код ошибки (возможные значения: V#16#A1, V#16#A2, V#16#A3, V#16#B1, V#16#B2, V#16#B3, V#16#B4)
OB85_PRIORITY	BYTE	Класс приоритета: 26 (значение по умолчанию для режима RUN) или 28 (режим STARTUP) В резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты с 0 по 2: Номер стойки Бит 3: 0= резервный CPU, 1=основной CPU Биты с 4 по 7: 1111
OB85_OB_NUMBR	BYTE	Номер OB (85)
OB85_RESERVED_1	BYTE	Резерв
OB85_RESERVED_2	BYTE	Резерв
OB85_RESERVED_3	INT	Резерв
OB85_ERR_EV_CLASS	BYTE	Класс события, вызвавшего ошибку
OB85_ERR_EV_NUM	BYTE	Номер события, вызвавшего ошибку
OB85_OB_PRIOR	BYTE	Класс приоритета OB, который был активен, когда произошла ошибка
OB85_OB_NUM	BYTE	Номер OB, который был активен, когда произошла ошибка
OB85_DATE_TIME	DATE_AND_TIME	Дата и время дня, когда вызывался OB



Если Вы хотите программировать OB85 в зависимости от возможных кодов ошибок, то рекомендуем Вам организовать локальные переменные следующим образом:

Переменная	Тип
OB85_EV_CLASS	BYTE
OB85_FLT_ID	BYTE
OB85_PRIORITY	BYTE
OB85_OB_NUMBR	BYTE
OB85_DKZ23	BYTE
OB85_RESERVED_2	BYTE
OB85_Z1	WORD
OB85_Z23	DWORD
OB85_DATE_TIME	DATE_AND_TIME

Переменные, изменяемые по сравнению со значением по умолчанию, имеют в зависимости от кода ошибки следующее значение:

- |              |           |                |   |
|--------------|-----------|----------------|---|
| • Код ошибки | V#16#A1   |                | В результате Вашей конфигурации, созданной с помощью STEP 7, Ваша программа или операционная система создает стартовое событие для OB, который не загружен в CPU. |
|              | V#16#A2   |                | В результате Вашей конфигурации, созданной с помощью STEP 7, Ваша программа или операционная система создает стартовое событие для OB, который не загружен в CPU. |
|              | OB85_Z1   |                | Соответствующая локальная переменная запрошенного OB, которая определяется OB85_Z23.  |
|              | OB85_Z23: | старшее слово: | Класс и номер события, вызывающего вызов OB.  |
|              |           | младшее слово: | Программный уровень и OB, активный во время ошибки.   |
- |              |          |  |   |
|--------------|----------|--|---|
| • Код ошибки | V#16#A3  |  | Ошибка при обращении операционной системы к модулю.<br>Идентификатор ошибки операционной системы.   |
|              | OB85_Z1: | старший байт:                                    | 1: встроенная функция   |
|              |          | младший байт:                                    | 2: таймер IEC   |
|              |          | 0: разрешающая способность при отсутствии ошибки | 3: ошибка защиты от записи  |
|              | OB85_Z23 | старшее слово:                                   | Номер блока   |
|              |          | младшее слово:                                   | Относительный адрес команды MC7, вызывающей ошибку. Тип блока может быть взят из OB85_DKZ23 (V#16#88: OB, V#16#8C: FC, V#16#8E: FB, V#16#8A: DB). |

- Код ошибки V#16#B1: Ошибка доступа к входам/выходам во время обновления всего образа процесса на входах.
- V#16#B2: Ошибка доступа к входам/выходам во время обновления всего образа процесса на выходах.
- OV85\_Z1: Зарезервировано для внутреннего использования CPU.
- OV85\_Z23: байт 0: Номер раздела образа процесса.
- байты 2,3: Номер байта входов/выходов, вызвавшего ошибку доступа к входам/выходам.

Вы получаете коды ошибки V#16#B1 и V#16#B2, если Вы сконфигурировали повторную сигнализацию ошибок доступа к входам/выходам для обновления таблицы образа процесса системой.

- Код ошибки V#16#B3: Ошибка доступа к входам/выходам при обновлении входной таблицы образа процесса (появление/уход).
- V#16#B4: Ошибка доступа к входам/выходам при передаче выходной таблицы образа процесса модулям вывода (появление/уход).
- OV85\_Z1: байт 0: Номер раздела образа процесса.
- OV85\_Z23: байты 2,3: Номер байта входов/выходов, вызвавшего ошибку доступа к входам/выходам.

Вы получаете коды ошибки V#16#B3 и V#16#B4, если Вы сконфигурировали сигнализацию ошибок доступа к входам/выходам при входе и выходе из состояния для обновления таблицы образа процесса системой.

## 1.21 Организационный блок неисправностей стоек (OB86)

### Описание

Операционная система CPU вызывает OB 86 в случаях, когда распознается неисправность стойки расширения, master-системы DP или станции у децентрализованной периферии (как при наступающем, так и при уходящем событии).

Если OB 86 не был запрограммирован, а обнаруживается такая ошибка, то CPU переходит в состояние STOP.

OB 86 можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Локальные данные OB неисправностей стоек

Следующая таблица описывает временные (TEMP) переменные OB неисправностей стоек. В качестве имен переменных взяты имена по умолчанию OB86.

Переменная	Тип	Описание
OB86_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#38: уходящее событие В#16#39: наступающее событие
OB86_FLT_ID	BYTE	Код ошибки: (возможные значения В#16#C1, В#16#C2, В#16#C3, В#16#C4, В#16#C5, В#16#C6, В#16#C7, В#16#C8)
OB86_PRIORITY	BYTE	Класс приоритета: 26 (значение по умолчанию для режима RUN) или 28 (режим STARTUP) На резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2:       Номер стойки Бит 3:            0=резервный CPU, 1=основной CPU Биты 4 – 7:       1111
OB86_OB_NUMBR	BYTE	Номер OB (86)
OB86_RESERVED_1	BYTE	Резерв
OB86_RESERVED_2	BYTE	Резерв
OB86_MDL_ADDR	WORD	Зависит от кода ошибки
OB86_RACKS_FLTD	Array [0 ..31] of BOOL	Зависит от кода ошибки
OB86_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

Если Вы желаете запрограммировать OB 86 в зависимости от кодов ошибок, то рекомендуется локальные переменные организовывать следующим образом:

Переменная	Тип
OB86_EV_CLASS	BYTE
OB86_FLT_ID	BYTE
OB86_PRIORITY	BYTE
OB86_OB_NUMBR	BYTE
OB86_RESERVED_1	BYTE
OB86_RESERVED_2	BYTE
OB86_MDL_ADDR	WORD
OB86_Z23	DWORD
OB86_DATE_TIME	DATE_AND_TIME

Переменные, содержимое которых зависит от кодов ошибок, имеют следующее значение:

- Код ошибки V#16#C1: Неисправность стойки расширения  
Логический базовый адрес IM
- OB86\_MDL\_ADDR:  
OB86\_Z23:
- Содержит по одному биту для каждой возможной стойки расширения:  
Бит:  
Бит 0: всегда 0  
Бит 1: 1-я стойка расширения  
:  
:  
Бит 21: 21-я стойка расширения  
Бит 22 – 29: всегда 0  
Бит 30: Неисправность по крайней мере одной стойки расширения в области SIMATIC S5  
Бит 31: всегда 0

**Значение:** когда происходит событие, стойки расширения, обусловившие вызов OB86, индицируются как неисправные (соответствующие им биты устанавливаются). Стойки расширения, вышедшие из строя ранее, больше не индицируются. Когда неисправность устраняется, в коде ошибки сообщается о стойках расширения, вновь ставших активными (соответствующие им биты устанавливаются).

- Код ошибки V#16#C2: Восстановление стойки расширения (с расхождением между ожидаемой и фактической конфигурацией)  
Логический базовый адрес IM  
OB86\_MDL\_ADDR:  
OB86\_Z23: Содержит один бит для каждой возможной стойки расширения, см. код ошибки V#16#C1.  
Значение установленного бита (для соответствующей стойки расширения)
  - имеются модули с неправильным идентификатором типа
  - отсутствуют сконфигурированные модули
  - хотя бы один модуль неисправен.
- Код ошибки V#16#C3: Децентрализованная периферия: неисправность master-системы. (Только наступающее событие ведет к старту OB 86 с кодом ошибки V#16#C3. Уходящее событие запускает OB86 с кодом ошибки V#16#C4 и классом события V#16#38. Восстановление любой slave-станции DP вызывает запуск OB86.)  
OB86\_MDL\_ADDR: Логический базовый адрес ведущие DP-устройства.

Системные и стандартные функции для S7-300 и S7-400  
A5E00261410-01

<p>OB86_Z23:</p>	<p>Идентификатор master-системы DP          Биты 0 – 7: резерв          Биты 8 – 15: идентификатор master-системы DP          Биты 16 – 31: резерв</p>
<p>• Код ошибки</p>	<p>V#16#C4: Выход из строя станции DP.          V#16#C5: Децентрализованная периферия: сбой станции DP.</p>
<p>OB86_MDL_ADDR:          OB86_Z23:</p>	<p>Логический базовый адрес ведущие DP-устройства.          Адрес неисправного ведомые DP-устройства:          Биты 0 – 7: номер станции DP          Биты 8 – 15: идентификатор master-системы DP          Биты 8 – 30: логический базовый адрес slave-устройства S7 или диагностический адрес стандартного ведомые DP-устройства          Бит 31: идентификатор ввода/вывода</p>
<p>• Код ошибки</p>	<p>V#16#C6: Восстановление стойки расширения, но ошибка при назначении параметров модуля</p>
<p>OB86_MDL_ADDR:          OB86_Z23:</p>	<p>Логический базовый адрес IM          Содержит один бит для каждой возможной стойки расширения:          Бит 0: всегда 0          Бит 1: 1-я стойка расширения          :          :          Бит 21: 21-я стойка расширения          Биты 22 – 30: резерв          Бит 31: всегда 0          Значение установленного бита (в соответствующей стойке расширения):          Существуют модули с неправильным идентификатором типа          Существуют модули с неправильными или незадаанными параметрами</p>
<p>• Код ошибки</p>	<p>V#16#C7: Восстановление станции DP, но ошибка при назначении параметров модуля</p>
<p>OB86_MDL_ADDR:          OB86_Z23:</p>	<p>Логический базовый адрес ведущие DP-устройства          Адрес неисправного ведомые DP-устройства:          Биты 0 – 7: номер станции DP          Биты 8 – 15: идентификатор master-системы DP          Биты 16 – 30: логический базовый адрес slave-устройства DP          Бит 31: идентификатор ввода/вывода</p>
<p>• Код ошибки</p>	<p>V#16#C8: Восстановление станции DP, однако имеется расхождение между сконфигурированной и фактической конфигурацией</p>
<p>OB86_MDL_ADDR:          OB86_Z23:</p>	<p>Логический базовый адрес ведущие DP-устройства          Адрес неисправного ведомые DP-устройства:          Биты 0 – 7: номер станции DP          Биты 8 – 15: идентификатор master-системы DP          Биты 16 – 30: логический базовый адрес slave-устройства DP          Бит 31: идентификатор ввода/вывода</p>

## 1.22 Организационный блок коммуникационных ошибок (OB87)

### Описание

Операционная система CPU вызывает OB 87 в случаях, когда наступает событие, которое было вызвано коммуникационной ошибкой.

Если OB 87 не был запрограммирован, то CPU при обнаружении такого события переходит в состояние STOP.

OB коммуникационных ошибок можно запретить или отложить и вновь разрешить с помощью SFC 39 – 42.

### Локальные данные OB87

Следующая таблица описывает временные (TEMP) переменные OB коммуникационных ошибок. В качестве имен переменных взяты имена по умолчанию OB87.

Переменная	Тип	Описание
OB87_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#35
OB87_FLT_ID	BYTE	Код ошибки: (возможные значения V#16#D2, V#16#D3, V#16#D4, V#16#D5, V#16#E1, V#16#E2, V#16#E3, V#16#E4, V#16#E5 или V#16#E6)
OB87_PRIORITY	BYTE	Класс приоритета: 26 (значение по умолчанию для режима RUN) или 28 (режим STARTUP) На резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2:       Номер стойки Бит 3:            0=резервный CPU, 1=основной CPU Биты 4 – 7:       1111
OB87_OB_NUMBR	BYTE	Номер OB (87)
OB87_RESERVED_1	BYTE	Резерв
OB87_RESERVED_2	BYTE	Резерв
OB87_RESERVED_3	WORD	Зависит от кода ошибки
OB87_RESERVED_4	DWORD	Зависит от кода ошибки
OB87_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB



## 1.23 ОВ обработки прерывания выполнения программы (ОВ 88)

### Описание

Операционная система CPU вызывает ОВ 88 после прерывания обработки программного блока. Возможные случаи применения этого прерывания:

- слишком велика глубина вложения синхронных ошибок
- слишком велика глубина вложения вызовов блоков (I-стек)
- ошибка при размещении локальных данных

Если ОВ 88 не запрограммирован и выполнение программного блока прервано, то CPU переходит в режим STOP (событие ID W#16#4570).

Если выполнение программного блока прервано в приоритетном классе 28, то CPU переходит в режим STOP.

Вы можете запретить (disable), задержать (delay) и разрешить (enable) ОВ обработки прерывания выполнения программы с помощью SFC 39 - SFC 42.

### Локальные данные ОВ обработки прерывания выполнения программы блока

Следующая таблица содержит временные переменные (TEMP) ОВ обработки прерывания выполнения программы блока. Имена по умолчанию для ОВ 88 выбраны как имена переменных.

Переменная	Тип данных	Описание
OB88_EV_CLASS	BYTE	Класс события и ID: V#16#35
OB88_SW_FLT	BYTE	Возможные значения кодов ошибок: <ul style="list-style-type: none"> <li>• V#16#73: слишком велика глубина вложения синхронных ошибок</li> <li>• V#16#75: слишком велика глубина вложения вызовов блоков (I-стек)</li> <li>• V#16#76: ошибка при размещении локальных данных</li> </ul>
OB88_PRIORITY	BYTE	• Приоритетный класс: 28
OB88_OB_NUMBR	BYTE	Номер ОВ (88)
OB88_BLK_TYPE	BYTE	Тип блока, в котором произошла ошибка: <ul style="list-style-type: none"> <li>• V#16#88: ОВ</li> <li>• V#16#8C: FC</li> <li>• V#16#8E: FB</li> <li>• V#16##00: источник прерывания не определен</li> </ul>
OB88_RESERVED_1	BYTE	Зарезервирован
OB88_FLT_PRIORITY	BYTE	Приоритетный класс ОВ, в котором ошибка
OB88_FLT_OB_NUMBR	BYTE	Номер ОВ, в котором произошла ошибка
OB88_BLK_NUM	WORD	Номер блока с MC7 инструкцией, в которой произошла ошибка
OB88_PRG_ADDR	WORD	Относительный адрес MC7 инструкции, в которой произошла ошибка
OB88_DATE_TIME	DATE_AND_TIME	Дата и TOD вызова ОВ



## 1.24 Организационный блок фонового режима (OB90)

### Описание

С помощью STEP 7 можно контролировать максимальное время цикла и гарантировать минимальное время цикла. Если время исполнения OB1, включая все вложенные прерывания и системные операции, оказывается меньше заданной вами минимальной длительности цикла, то операционная система реагирует следующим образом:

- Она вызывает OB фонового режима (если он имеется в CPU).
- Она задерживает следующий запуск OB1 (если в CPU отсутствует OB 90).

### Принцип действия OB90

OB90 имеет самый низкий приоритет среди всех остальных OB. Он прерывается любой системной операцией и любым прерыванием и возобновляется только в том случае, если заданное минимальное время цикла еще не достигнуто. Исключение составляет обработка SFC и SFB, которые были запущены из OB90. Они исполняются с приоритетом OB1 и поэтому не прерываются OB1. Контроля длительности OB 90 не ведется.

Программа пользователя в OB90 обрабатывается, начиная с первой команды, в следующих ситуациях:

- после теплого, холодного или горячего рестарта
- после удаления блока, исполняемого в OB90 (с помощью STEP 7)
- после загрузки OB 90 в CPU в режиме RUN
- по окончании фонового цикла.

---

### Примечание

В случае конфигураций, в которых нет большой разницы между минимальным временем цикла и контрольным временем цикла, вызовы SFC и SFB в фоновом OB могут привести к непредусмотренному превышению времени цикла.

---

**Локальные данные OB90**

Следующая таблица описывает временные (TEMP) OB 90. В качестве имен переменных взяты имена по умолчанию OB90.

Переменная	Тип данных	Описание
OB90_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#11: активен
OB90_STRT_INF	BYTE	V#16#91: теплый рестарт/холодный рестарт/ горячий рестарт V#16#92: блок удален V#16#93: загрузка OB 90 в CPU в режиме RUN V#16#95: окончание фонового цикла
OB90_PRIORITY	BYTE	Класс приоритета: 29 (соответствует приоритету 0.29)
OB90_OB_NUMBR	BYTE	Номер OB (90)
OB90_RESERVED_1	BYTE	резерв
OB90_RESERVED_2	BYTE	резерв
OB90_RESERVED_3	INT	резерв
OB90_RESERVED_4	INT	резерв
OB90_RESERVED_5	INT	резерв
OB90_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB

## 1.25 Организационные блоки запуска (OB100, OB101 и OB102)

### Типы запуска

Различают следующие типы запуска:

- Горячий рестарт (нет у S7-300 и S7-400H)
- Теплый рестарт
- Холодный рестарт

В следующей таблице Вы можете видеть, какой OB вызывается операционной системой во время запуска.

Тип запуска	Соответствующий OB
Горячий рестарт	OB101
Теплый рестарт	OB100
Холодный рестарт	OB102

За более подробной информацией о типах запуска обратитесь к руководствам "Programming and Hardware Configuration with STEP 7 [Программирование и конфигурирование аппаратных средств с помощью STEP 7]" и "S7-400H Programmable Controller [Программируемый контроллер S7-400H]".

### Стартовые события для запуска

CPU выполняет запуск в следующих случаях:

- после включения питания
- всякий раз, когда Вы переводите переключатель режимов работы из STOP в RUN-P
- после запроса, использующего коммуникационную функцию (посредством команды меню из устройства программирования или вызова коммуникационных функциональных блоков 19 "START" или 21 "RESUME" в другом CPU)
- синхронизация в многопроцессорном режиме
- в H-системе после соединения (только в резервном CPU).

В зависимости от стартового события, конкретного CPU и его параметров вызывается соответствующий OB запуска (OB100, OB101 или OB102).  
Посредством подходящего программирования Вы можете настраивать определенные параметры Вашей циклической программы (исключение: в H-системе, когда подключен резервный CPU, в резервном CPU имеет место запуск, но OB запуска не вызывается).

**Локальные данные для OB запуска**

Следующая таблица описывает временные (TEMP) переменные для стартовых OB. Имена переменных являются заданными по умолчанию именами OB100.

Переменная	Тип	Характеристика
OB10x_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#13: активный
OB10x_STRTUP	BYTE	Запрос запуска: В#16#81: Ручной теплый рестарт В#16#82: Автоматический теплый рестарт В#16#83: Запрос ручного горячего рестарта В#16#84: Запрос автоматического горячего рестарта В#16#85: Запрос ручного холодного рестарта В#16#86: Запрос автоматического холодного рестарта В#16#87: Главный: Запрос ручного холодного рестарта В#16#88: Главный: Запрос автоматического холодного рестарта В#16#8A: Главный: Запрос ручного теплого рестарта В#16#8B: Главный: Запрос автоматического теплого рестарта В#16#8C: Резервный: Запрос ручного рестарта В#16#8D: Резервный: Запрос автоматического рестарта
OB10x_PRIORITY	BYTE	Класс приоритета: 27
OB10x_OB_NUMBR	BYTE	Номер OB (100, 101 или 102)
OB10x_RESERVED_1	BYTE	Резерв
OB10x_RESERVED_2	BYTE	Резерв
OB10x_STOP	WORD	Номер события, вызвавшего останов CPU
OB10x_STRT_INFO	DWORD	Дополнительная информация о текущем запуске
OB10x_DATE_TIME	DATE_AND_TIME	Дата и время дня, когда вызывался OB

Следующая таблица показывает переменные OB100\_STR\_INFO и OB101\_STR\_INFO.

№ бита	Значение	Возможные двоичные значения	Объяснение
31 - 24	Стартовая информация	0000 xxxx 0100 xxxx 1000 xxxx 0001 xxxx 0010 xxxx xxxx xxx0 xxxx xxx1 xxxx xx0x xxxx xx1x	Номер стойки 0 (только H CPU) Номер стойки 1 (только H CPU) Номер стойки 2 (только H CPU) Многопроцессорный режим (только S7-400) Работа более одного CPU в сегментированной стойке (только S7-400) Нет различия между ожидаемой и фактической конфигурацией (только S7-300) Есть различие между ожидаемой и фактической конфигурацией (только S7-300) Нет различия между ожидаемой и фактической конфигурацией Есть различие между ожидаемой и фактической конфигурацией

Системные и стандартные функции для S7-300 и S7-400  
A5E00261410-01

№ бита	Значение	Возможные двоичные значения	Объяснение
		xxxx x0xx xxxx x1xx xxxx 0xxx  xxxx 1xxx	Не Н CPU Н CPU Часы для отметки времени при последнем включении питания без резервного батарейного питания Часы для отметки времени при последнем включении питания с резервным батарейным питанием
23 - 16	Запуск только что завершился	0000 0001  0000 0011  0000 0100	Теплый рестарт в многопроцессорной системе обработки без изменения установки в CPU согласно назначению параметров (только S7-400) Рестарт (теплый), запущенный с помощью переключателя режимов работы Рестарт (теплый), запущенный командой через MPI
		0000 0101  0000 0011  0000 1000  0000 1010  0000 1011  0000 1100  0001 0000  0001 0001  0001 0011  0001 0100  0010 0000  0010 0001  0010 0011  0010 0100  1010 0000	Холодный рестарт в многопроцессорной системе обработки без изменения установки в CPU согласно назначению параметров (только S7-400) Холодный рестарт, запущенный с помощью переключателя режимов работы Холодный рестарт, запущенный командой через MPI Горячий рестарт в многопроцессорной системе обработки без изменения установки в CPU согласно назначению параметров (только S7-400) Горячий рестарт, запущенный с помощью переключателя режимов работы (только S7-400) Горячий рестарт, запущенный командой через MPI (только S7-400) Автоматический рестарт (теплый) после включения питания при наличии резервного батарейного питания Автоматический холодный рестарт после включения питания при наличии резервного батарейного питания Рестарт (теплый), запущенный с помощью переключателя режимов работы; последнее включение питания при наличии резервного батарейного питания Рестарт (теплый), запущенный командой через MPI; последнее включение питания при наличии резервного батарейного питания Автоматический рестарт (теплый) после включения питания при наличии резервного батарейного питания (со сбросом памяти системой) Автоматический холодный рестарт после включения питания при наличии резервного батарейного питания (со сбросом памяти системой) Рестарт (теплый), запущенный с помощью переключателя режимов работы; последнее включение питания без резервного батарейного питания Рестарт (теплый), запущенный командой через MPI; последнее включение питания без резервного батарейного питания Автоматический горячий рестарт после включения питания при наличии резервного батарейного питания согласно назначению параметров (только S7-400)

№ бита	Значение	Возможные двоичные значения	Объяснение
15 - 12	Допустимость автоматического запуска	0000	Автоматический запуск запрещен, требуется сброс памяти
		0001	Автоматический запуск запрещен, необходимы изменения параметров и т.д.
		0111	Автоматический запуск (теплый) разрешен
		1111	Автоматический запуск (теплый/горячий) разрешен (только S7-400)
11 - 8	Допустимость ручного запуска	0000	Запуск запрещен, запрошен сброс памяти
		0001	Запуск запрещен, запрошены изменения параметров и т.д.
		0111	Запуск (теплый) разрешен
		1111	Запуск (теплый/горячий) разрешен (только S7-400)
7 - 0	Последнее допустимое вмешательство или установка автоматического запуска при включении питания	0000 0000	Нет запуска
		0000 0001	Теплый рестарт в многопроцессорной системе без изменения установки в CPU согласно назначению параметров (только S7-400)
		0000 0011	Рестарт (теплый), запущенный с помощью переключателя
		0000 0100	Рестарт (теплый), запущенный командой через MPI
		0000 1010	Горячий рестарт в многопроцессорной системе без изменения установки в CPU согласно назначению параметров (только S7-400)
		0000 1011	Горячий рестарт, запущенный с помощью переключателя режимов работы (только S7-400)
		0000 1100	Горячий рестарт, запущенный командой через MPI (только S7-400)
		0001 0000	Автоматический рестарт (теплый) после включения питания при наличии резервного батарейного питания
		0001 0011	Рестарт (теплый), запущенный командой через MPI; последнее включение питания при наличии резервного батарейного питания
		0001 0100	Рестарт (теплый), запущенный командой через MPI; последнее включение питания при наличии резервного батарейного питания
		0010 0000	Автоматический рестарт (теплый) после включения питания при наличии резервного батарейного питания (со сбросом памяти системой)
		0010 0011	Рестарт (теплый), запущенный с помощью переключателя режимов работы; последнее включение питания без резервного батарейного питания
		0010 0100	Рестарт (теплый), запущенный командой через MPI; последнее включение питания без резервного батарейного питания
1010 0000	Автоматический горячий рестарт после включения питания при наличии резервного батарейного питания согласно назначению параметров (только S7-400)		

## 1.26 Организационный блок ошибок программирования (OB121)

### Описание

Операционная система CPU вызывает OB 121 в тех случаях, когда происходит событие, вызванное ошибкой, связанной с обработкой программы. Если, например, Вы вызвали в своей программе блок, который не был загружен в CPU, то вызывается OB 121.

### Принцип работы OB ошибок программирования

OB121 исполняется в том же классе приоритета, что и прерванный блок. Если OB121 не был запрограммирован, то CPU переходит в состояние STOP. S7 предоставляет в распоряжение следующие SFC, с помощью которых Вы можете маскировать и демаскировать стартовые события OB 121 во время обработки Вашей программы:

- SFC36 (MSK\_FLT): маскирует определенные коды ошибок
- SFC37 (DMSK\_FLT): демаскирует коды ошибок, которые были замаскированы с помощью SFC36
- SFC38 (READ\_ERR): читает регистр ошибок

**Локальные данные OB ошибок программирования**

Следующая таблица описывает временные (TEMP) переменные OB ошибок программирования. В качестве имен переменных взяты имена по умолчанию OB121.

Переменная	Тип	Описание
OB121_EV_CLASS	BYTE	Класс события и идентификаторы: В#16#25
OB121_SW_FLT	BYTE	Код ошибки : (возможные значения В#16#21, В#16#22, В#16#23, В#16#24, В#16#25, В#16#26, В#16#27, В#16#28, В#16#29, В#16#30, В#16#31, В#16#32, В#16#33, В#16#34, В#16#35, В#16#3А, В#16#3С, В#16#3D, В#16#3Е или В#16#3F)
OB121_PRIORITY	BYTE	Класс приоритета: класс приоритета OB, в котором произошла ошибка На резервном CPU Н-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2:       Номер стойки Бит 3:            0=резервный CPU, 1=основной CPU Биты 4 – 7:       1111
OB121_OB_NUMBR	BYTE	Номер OB (121)
OB121_BLK_TYPE	BYTE	Тип блока, в котором произошла ошибка (для S7-300 сюда не заносится никакого действительного значения): В#16#88: OB, В#16#8А: DB, В#16#8С: FC, В#16#8Е: FB
OB121_RESERVED_1	BYTE	резерв
OB121_FLT_REG	WORD	Источник ошибки (зависит от кода ошибки). Например: Регистр, в котором возникла ошибка преобразования Неправильный адрес (для ошибки чтения/записи) Неправильный номер таймера, счетчика или блока Неправильный идентификатор области памяти
OB121_BLK_NUM	WORD	Номер блока с командой MC7, вызвавшей ошибку (для S7-300 сюда не заносится никакого действительного номера)
OB121_PRG_ADDR	WORD	Относительный адрес команды MC7, вызвавшей ошибку (для S7-300 сюда не заносится никакого действительного значения)
OB121_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван OB



Переменные, зависящие от кодов ошибок, имеют следующие значения:

- Код ошибки  
OB121\_FLT\_REG: V#16#21: Ошибка преобразования в BCD-код.  
Идентификатор соответствующего регистра  
(W#16#0000: аккумулятор 1)
- Код ошибки  
V#16#22: Ошибка длины области при чтении  
V#16#23: Ошибка длины области при записи  
V#16#28: Обращение для чтения к байту, слову или двойному слову  
с указателем, битовый адрес которого не равен нулю.  
V#16#29: Обращение для записи к байту, слову или двойному слову  
с указателем, битовый адрес которого не равен нулю.  
OB121\_FLT\_REG: Ошибочный байтовый адрес. Область данных и тип  
доступа можно считать из OB121\_RESERVED\_1.  
OB121\_RESERVED\_1: Биты 7 – 4 тип доступа.  
0: битовый доступ, 1: байтовый доступ, 2: доступ к слову,  
3: доступ к двойному слову  
Биты 3 – 0 область памяти:  
0: область периферии,  
1: таблица образа процесса на входах,  
2: таблица образа процесса на выходах,  
3: память с побитовым доступом (меркеры), 4: глобальный  
DB, 5: экземпляр DB, 6: собственные локальные данные, 7:  
локальные данные вызывающего блока
- Код ошибки  
OB121\_FLT\_REG: V#16#24: Ошибка области при чтении  
V#16#25: Ошибка области при записи  
Содержит в младшем байте идентификатор недопустимой  
области (V#16#86 из области собственных локальных  
данных)
- Код ошибки  
OB121\_FLT\_REG: V#16#26: Ошибка номера таймера  
V#16#27: Ошибка номера счетчика  
Недопустимый номер
- Код ошибки  
V#16#30: Попытка записи в защищенный от записи глобальный DB  
V#16#31: Попытка записи в защищенный от записи экземпляр DB  
V#16#32: Ошибка номера DB при обращении к глобальному DB  
V#16#33: Ошибка номера DB при обращении к экземпляру DB  
OB121\_FLT\_REG: Недопустимый номер DB
- Код ошибки  
V#16#34: Ошибка номера FC при вызове FC  
V#16#35: Ошибка номера FB при вызове FB  
V#16#3A: Обращение к незагруженному DB; номер DB находится в  
допустимом диапазоне  
V#16#3C: Обращение к незагруженной FC; номер FC находится в  
допустимом диапазоне  
V#16#3D: Обращение к незагруженной SFC; номер SFC лежит в  
допустимом диапазоне  
V#16#3E: Обращение к незагруженному FB; номер FB лежит в  
допустимом диапазоне  
V#16#3F: Обращение к незагруженному SFB; номер SFB лежит в  
допустимом диапазоне  
OB121\_FLT\_REG: Недопустимый номер

## 1.27 Организационный блок ошибок доступа к периферии (OB122)

### Описание

Операционная система CPU вызывает OB 122 в случае, если при обращении к данным какого-либо модуля происходит ошибка. Если, например, CPU распознает ошибку чтения при обращении к данным модуля ввода/вывода, то операционная система вызывает OB122.

### Принцип работы OB ошибок доступа к периферии

OB122 выполняется в том же классе приоритета, что и прерванный блок. Если OB 122 не запрограммирован, то CPU переходит из RUN в STOP.

S7 предоставляет следующие SFC, с помощью которых можно маскировать и демаскировать стартовые события OB 122 во время исполнения Вашей программы:

- SFC36 (MSK\_FLT): маскирует определенные коды ошибок
- SFC37 (DMSK\_FLT): демаскирует коды ошибок, замаскированные с помощью SFC36
- SFC38 (READ\_ERR): читает регистр ошибок

### Локальные данные OB ошибок доступа к периферии

Следующая таблица описывает временные (TEMP) переменные OB ошибок доступа к периферии. В качестве имен переменных взяты имена по умолчанию OB122.

Переменная	Тип	Описание
OB122_EV_CLASS	BYTE	Класс события и идентификаторы: V#16#29
OB122_SW_FLT	BYTE	Код ошибки: V#16#42 Для S7-300 и CPU 417: ошибка доступа к периферии при чтении Для всех остальных CPU S7-400: ошибка при первом обращении для чтения после появления ошибки V#16#43 Для S7-300 и CPU 417: ошибка доступа к периферии при записи Для всех остальных CPU S7-400: ошибка при первом обращении для записи после появления ошибки V#16#44 (только для S7-400, кроме CPU 417) ошибка при n-ом обращении для чтения (n > 1) после появления ошибки V#16#45 (только для S7-400, кроме CPU 417) ошибка при n-ом обращении для записи (n > 1) после появления ошибки

Переменная	Тип	Описание
OB122_PRIORITY	BYTE	Класс приоритета: Класс приоритета ОБ, в котором произошла ошибка На резервном CPU H-системы в режиме LINK-UP, STARTUP или UPDATE: Биты 0 – 2:       Номер стойки Бит 3:             0=резервный CPU, 1=основной CPU Биты 4 – 7:       1111
OB122_OB_NUMBR	BYTE	Номер ОБ (122)
OB122_BLK_TYPE	BYTE	Тип блока, в котором произошла ошибка (В#16#88: ОБ, В#16#8А: DB, В#16#8С: FC, В#16#8Е: FB) (для S7-300 здесь не вводится никакого действительного значения)
OB122_MEM_AREA	BYTE	Область памяти и тип доступа: Биты 7 – 4 тип доступа: 0: битовый доступ, 1: байтовый доступ, 2: доступ к слову, 3: доступ к двойному слову Биты 3 – 0 область памяти: 0: область периферии, 1: образ процесса на входах, 2: образ процесса на выходах
OB122_MEM_ADDR	WORD	Адрес в памяти, на котором произошла ошибка
OB122_BLK_NUM	WORD	Номер блока с командой MC7, вызвавшей ошибку (для S7-300 здесь не вводится никакого действительного номера)
OB122_PRG_ADDR	WORD	Относительный адрес команды MC7, вызвавшей ошибку (для S7-300 здесь не вводится никакого действительного значения)
OB122_DATE_TIME	DATE_AND_TIME	Дата и время, когда был вызван ОБ



## **2 Общие параметры для SFC**

## 2.1 Оценка ошибок с помощью выходного параметра RET\_VAL

### Виды информации об ошибках

Системная функция (SFC), исполняемая в Вашей пользовательской программе показывает, смог ли CPU успешно выполнить функцию SFC. Вы можете получить информацию о любых произошедших ошибках двумя способами:

- в бите BR слова состояния
- в выходном параметре RET\_VAL (возвращаемое значение)

---

#### Примечание

Перед анализом выходных параметров, относящихся к SFC, Вы всегда должны выполнять следующие шаги:

- Сначала проанализируйте бит BR слова состояния.
- Затем проверьте выходной параметр RET\_VAL.

Если бит BR показывает, что произошла ошибка, или если RET\_VAL содержит код общей ошибки, то Вам нельзя использовать выходной параметр SFC!

---

### Информация об ошибках в возвращаемом значении

Системная функция (SFC) через значение "0" бита двоичного результата (BR) слова состояния показывает, что при исполнении функции произошла ошибка. Некоторые системные функции предоставляют в распоряжение дополнительный код ошибки на выходе, известном как возвращаемое значение (RET\_VAL). Если в выходной параметр RET\_VAL вводится общая ошибка (объяснение см. ниже), то она отображается только посредством значения "0" бита BR слова состояния.

Возвращаемое значение относится к типу данных "целое число" (INT). Отношение возвращаемого значения к значению "0" показывает, появилась ли при исполнении функции ошибка.

CPU исполняет SFC	BR	Возвращаемое значение	Знак целого числа
с ошибкой (ами)	0	меньше, чем "0"	отрицательный (знаковый бит равен "1")
без ошибки	1	больше или равно "0"	положительный (знаковый бит равен "0")

### Реакция на информацию об ошибках

Имеются следующие два различных типа кодов ошибки в RET\_VAL:

- общий код ошибки, который могут выдавать все системные функции, и
- специфический код ошибки, который SFC может выдавать в зависимости от своей конкретной функции.

Вы можете написать свою программу таким образом, что она будет реагировать на ошибки, возникающие при исполнении системной функции. Этим Вы препятствуете возникновению последующих ошибок, являющихся результатом первой ошибки.

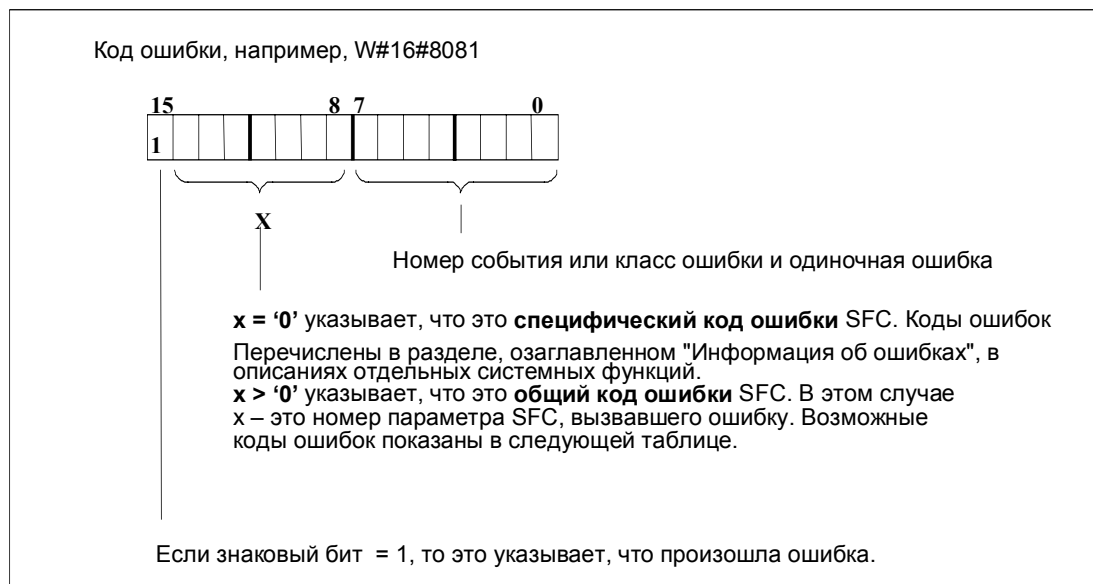
### Общая и специфическая информация об ошибках

Возвращаемое значение (RET\_VAL) системной функции предоставляет в распоряжение один из двух следующих типов кодов ошибки:

- Общий код ошибки, относящийся к ошибкам, которые могут возникнуть в любой системной функции.
- Специфический код ошибки, который относится только к конкретной системной функции.

Хотя тип данных выходного параметра RET\_VAL целый (INT), коды ошибок системных функций группируются в соответствии с шестнадцатеричными значениями. Если Вы хотите проанализировать возвращаемое значение и сравнить это значение с кодами ошибок, которые приведены в данном руководстве, то отобразите код ошибки в шестнадцатеричном формате.

Следующий рисунок показывает структуру кода ошибки системной функции в шестнадцатеричном формате.

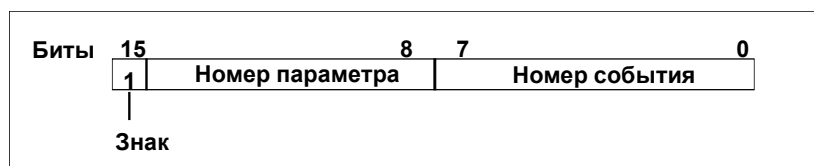


### Общая информация об ошибках

Общий код ошибки показывает ошибки, которые могут встречаться во всех системных функциях. Общий код ошибки состоит из двух следующих номеров:

- Номер параметра в диапазоне от 1 до 111, где 1 указывает на первый параметр, 2 - на второй параметр SFC и т.д.
- Номер события в диапазоне от 0 до 127. Номер события указывает на то, что произошла синхронная ошибка.

В следующей таблице перечислены коды для общих ошибок и объяснение каждой ошибки.



#### Примечание

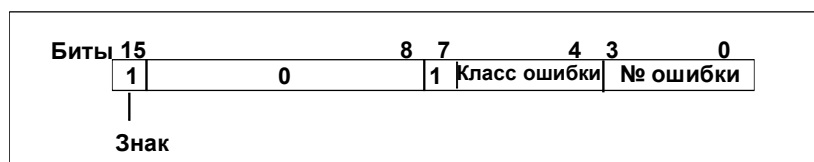
Если в RET\_VAL был записан общий код ошибки, то возможны следующие ситуации:

- Возможно, была запущена или уже выполнена операция, относящаяся к SFC.
- Возможно, что при выполнении этой операции произошла специфическая ошибка SFC. Однако в результате общей ошибки, произошедшей позднее, специфическая ошибка не могла больше отображаться.

### Специфическая информация об ошибках

Некоторые системные функции (SFC) обладают возвращаемым значением, которое предоставляет в распоряжение специфический код ошибки. Этот код ошибки показывает, что во время обработки функции появилась ошибка, которая относится к определенной системной функции (см. рисунок). Специфический код ошибки состоит из двух следующих чисел:

- Класс ошибки от 0 до 7.
- Номер ошибки от 0 до 15.





**Общие коды ошибок**

Следующая таблица объясняет общие коды возвращаемого значения. Код ошибки отображается в шестнадцатеричном формате. Буква x в каждом кодовом номере служит только как символ-заполнитель и представляет номер параметра системной функции, вызвавшего ошибку.

Код ошибки (W#16#...)	Объяснение
8x7F	Внутренняя ошибка Этот код ошибки указывает на внутреннюю ошибку в параметре x. Эта ошибка была вызвана не пользователем и не может быть им устранена.
8x22 8x23	Ошибка длины области при чтении параметра. Ошибка длины области при записи параметра. Этот код ошибки показывает, что параметр x полностью или частично находится за пределами диапазона адресов или что длина битового поля для параметра типа ANY не кратна 8.
8x24 8x25	Ошибка области при чтении параметра Ошибка области при записи параметра. Этот код ошибки показывает, что параметр x находится в области, которая недопустима для системной функции. За информацией о недопустимых областях обращайтесь к описаниям отдельных функций.
8x26	Параметр содержит слишком большой номер таймера. Этот код ошибки показывает, что таймер, заданный в параметре x, не существует.
8x27	Параметр содержит слишком большой номер счетчика (ошибка номера счетчика). Этот код ошибки показывает, что счетчик, указанный в параметре x, не существует.
8x28 8x29	Ошибка выравнивания при чтении параметра. Ошибка выравнивания при записи параметра. Этот код ошибки показывает, что ссылка на параметр x является битовым адресом, не равным 0.
8x30 8x31	Параметр находится в защищенном от записи глобальном DB. Параметр находится в защищенном от записи экземплярном DB. Этот код ошибки показывает, что параметр x находится в защищенном от записи блоке данных. Если этот блок данных был открыт самой системной функцией, то системная функция всегда возвращает значение W#16#8x30.
8x32 8x34 8x35	Параметр содержит слишком большой номер DB (ошибка номера DB). Параметр содержит слишком большой номер FC (ошибка номера FC). Параметр содержит слишком большой номер FB (ошибка номера FB). Этот код ошибки показывает, что параметр x содержит номер блока, который больше, чем максимально допустимый номер блока.
8x3A 8x3C 8x3E	Параметр содержит номер незагруженного DB. Параметр содержит номер незагруженного FC. Параметр содержит номер незагруженного FB.
8x42 8x43	Имела место ошибка доступа в то время, когда система пыталась считать параметр из периферийной области входов. Имела место ошибка доступа в то время, когда система пыталась считать параметр из периферийной области выходов.
8x44 8x45	Ошибка при n-ом ( $n > 1$ ) обращении для чтения после появления ошибки. Ошибка при n-ом ( $n > 1$ ) обращении для записи после появления ошибки. Этот код ошибки показывает, что получен отказ в доступе к требуемому параметру.

## 2.2 Значение параметров REQ, RET\_VAL и BUSY для асинхронных SFC

### Асинхронные SFC

Асинхронно работающие SFC – это SFC, которые вызываются более одного раза до того, как они завершат выполнение своих функций. Следующие SFC всегда выполняются асинхронно или в определенных ситуациях:

- SFC7 "DP\_PRAL"
- SFC11 "DPSYC\_FR"
- SFC12 "D\_ACT\_DP"
- SFC13 "DPNRM\_DG"
- SFC51 "RDSYSST"
- SFC55 "WR\_PARM"
- SFC56 "WR\_DPARAM"
- SFC57 "PARM\_MOD"
- SFC58 "WR\_REC"
- SFC59 "RD\_REC"
- SFC65 "X\_SEND"
- SFC67 "X\_GET"
- SFC68 "X\_PUT"
- SFC69 "X\_ABORT"
- SFC72 "I\_GET"
- SFC73 "I\_PUT"
- SFC74 "I\_ABORT"
- SFC82 "CREA\_DBL"
- SFC83 "READ\_DBL"
- SFC84 "WRIT\_DBL"
- SFC90 "H\_CTRL"
- SFC102 "RD\_DPARA"

### Идентификация задания

Если Вы запускаете аппаратное прерывание, выдаете команды управления для ведомых DP-устройств, запускаете передачу данных или прерываете неконфигурированное соединение с помощью одной из вышеперечисленных SFC, а затем вновь вызываете ту же SFC до завершения текущего задания, то реакция SFC будет зависеть от того, включает ли в себя второй вызов то же самое задание.

Следующая таблица объясняет, какие входные параметры определяют задание для каждой из этих функций SFC. Если эти параметры совпадают с параметрами еще не завершенного задания, то этот вызов SFC считается продолжающимся (follow-on) вызовом.

SFC	Задание идентифицируется посредством...
7 "DP_PRAL"	IOID, LADDR
11 "DPSYC_FR"	LADDR, GROUP, MODE
13 "DPNRM_DG"	LADDR
51 "RDSYSST"	SSL_ID, INDEX
55 "WR_PARM"	IOID, LADDR, RECNUM
56 "WR_DPARM"	IOID, LADDR, RECNUM
57 "PARM_MOD"	IOID, LADDR
58 "WR_REC"	IOID, LADDR, RECNUM
59 "RD_REC"	IOID, LADDR, RECNUM
65 "X_SEND"	DEST_ID, REQ_ID
67 "X_GET"	DEST_ID, VAR_ADDR
68 "X_PUT"	DEST_ID, VAR_ADDR
69 "X_ABORT"	DEST_ID
72 "I_GET"	IOID, LADDR, VAR_ADDR
73 "I_PUT"	IOID, LADDR, VAR_ADDR
74 "I_ABORT"	IOID, LADDR
82 "CREA_DBL"	LOW_LIMIT, UP_LIMIT, COUNT, ATTRIB, SRCBLK
83 "READ_DBL"	SRCBLK, DSTBLK
84 "WRIT_DBL"	SRCBLK, DSTBLK
90 "H_CTRL"	MODE, SUBMODE
102 "RD_DPARA"	LADDR, RECNUM

### Входной параметр REQ

Входной параметр REQ (request [запрос]) используется исключительно для запуска задания:

- Если Вы вызываете SFC для выполнения задания, которое в данный момент не активно, то задание запускается с помощью REQ = 1 (ситуация 1).
- Если определенное задание было запущено и еще не завершено, а Вы вызываете SFC вновь, чтобы выполнить то же самое задание (например, в ОВ циклических прерываний), то SFC не анализирует REQ (ситуация 2).

## Выходные параметры RET\_VAL и BUSY

С помощью выходных параметров RET\_VAL и BUSY отображается состояние исполнения.

- В ситуации 1 (первый вызов при REQ=1) в RET\_VAL вводится W#16#7001, если системные ресурсы свободны и входные параметры правильны. Тогда устанавливается BUSY. Если требуемые системные в данный момент используются или входные параметры содержат ошибки, то в RET\_VAL вносится соответствующий код ошибки, а BUSY имеет значение 0.

В ситуации 2 (вызов, когда активно то же самое задание) в RET\_VAL вносится W#16#7002 (это предупреждение о том, что это задание еще обрабатывается), а BUSY устанавливается.

- Следующее относится к последнему вызову задания:
  - Если не произошло ошибки, то с помощью SFC13 "DPNRM\_DG," SFC67 "X\_GET" и SFC72 "I\_GET" в RET\_VAL в виде положительного числа байтов вводится количество передаваемых данных. Тогда BUSY имеет значение 0. Если происходит ошибка, то RET\_VAL содержит информацию об этой ошибке, а BUSY 0.
  - Если не произошло ошибки, то с помощью SFC59 "RD\_REC" в RET\_VAL вводится размер записи данных в байтах или 0. В этом случае BUSY имеет значение 0. Если происходит ошибка, то в RET\_VAL вводится код ошибки, а BUSY имеет значение 0.
  - У всех остальных SFC, если задание было выполнено без ошибок, то в RET\_VAL вводится 0, а BUSY имеет значение 0. Если происходит ошибка, то в RET\_VAL вводится код ошибки, а BUSY имеет значение 0.

---

### Примечание

Если первый и последний вызов приходят одновременно, то реакция для RET\_VAL и BUSY будет такой же, как описано для последнего вызова.

---

## Обзор

Следующая таблица дает обзор описанных выше соотношений. В частности, она показывает возможные значения выходных параметров, если выполнение задания не завершено после вызова SFC.

---

### Примечание

После каждого вызова Вы должны проанализировать в своей программе соответствующие выходные параметры.

---

Связь между вызовом, REQ, RET\_VAL и BUSY во время выполнения задания.

Номер вызова	Тип вызова	REQ	RET_VAL	BUSY
1	Первый вызов	1	W#16#7001	1
			Код ошибки	0
от 2 до (n - 1)	Промежуточный вызов	Не имеет значения	W#16#7002	1
N	Последний вызов	Не имеет значения	W#16#0000 (исключения: SFC59 "RD_REC", если целевая область больше, чем передаваемая запись данных, и SFC13 "DPNRM_DG," SFC67 "X_GET" и SFC72 "I_GET", если не было ошибок)	0
			Код ошибки, если были ошибки	0



### **3 Функции копирования и манипулирования блоками**

### 3.1 Копирование переменных с помощью SFC20 "BLKMOV"

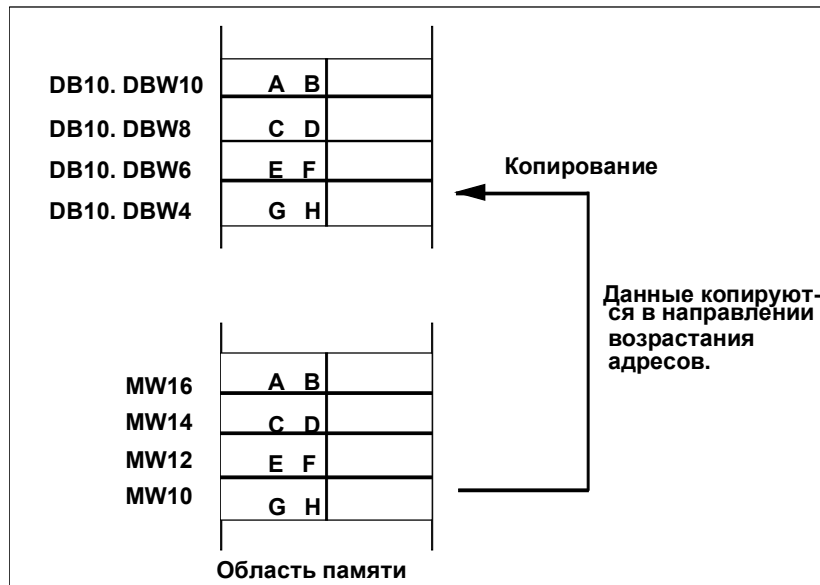
#### Описание

SFC20 "BLKMOV" (block move [перемещение блока]) используется для копирования содержимого области памяти (= исходной области) в другую область памяти (= целевую область).

С помощью SFC20 "BLKMOV" можно копировать любые области памяти, кроме:

- следующих типов блоков: FB, SFB, FC, SFC, OB, SDB,
- счетчиков,
- таймеров,
- областей памяти периферийных входов/выходов.

Исходные параметры могут быть включены в загружаемую память (область памяти блока, который не имеет значения для исполнения программы) (DB, скомпилированного с ключевым словом UNLINKED).





**Возможность прерывания**

Пока исходная область массив не является частью блока данных, который существует только в загрузочной памяти, нет ограничений на глубину вложения.

Если, однако, SFC20 прерывается при копировании из DB, не существенного для исполнения программы, то исполнение SFC20 более не может быть вложенным.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SRCBLK	INPUT	ANY	I, Q, M, D, L	Указывает области памяти, которая должна копироваться (исходная область). Массивы, относящиеся к типу данных STRING, недопустимы.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время исполнения функции происходит ошибка, то возвращаемое значение содержит код ошибки.
DSTBLK	OUTPUT	ANY	I, Q, M, D, L	Указывает область памяти, в которую будут копироваться данные (целевая область). Массивы, относящиеся к типу данных STRING, недопустимы.

**Примечание**

Исходная и целевая области не должны перекрываться. Если заданная целевая область больше, чем исходная, то функция копирует лишь столько данных, сколько содержится в исходной области.

Если заданная целевая область меньше, чем исходная, то функция копирует лишь столько данных, сколько может быть записано в целевую область.

Если указатель ANY (источника или адресата) имеет тип данных BOOL, то заданная длина должна делиться на 8; в противном случае SFC не будет выполняться.

Если указатель ANY имеет тип данных STRING, то заданная длина должна быть равна 1.

**Особенности:**

Если несвязанный блок данных копируется в ОЗУ с помощью SFC 20 BLKMOV и одновременно загружается, например, посредством PG, то выполнение функции SFC может вызвать задержку на отрезок времени длительностью до единиц миллисекунд. Это приводит к увеличению длительности цикла ОВ, и может нарушать текущий цикл мониторинга. Избегайте загрузки блока в CPU во время копирования блока с помощью SFC 20.

**Информация об ошибках**

<b>Код ошибки (W#16#...)</b>	<b>Объяснение</b>
0000	Нет ошибок
8091	Превышена глубина вложения
8092	SFC20 "BLKMOV" не может быть выполнен из-за обращения к неподходящему блоку. Используйте для операции перемещения SFC 83.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 3.2 Непрерываемое копирование переменных с помощью SFC81 "UBLKMOV"

### Описание

С помощью SFC81 "UBLKMOV" (uninterruptable block move [непрерываемое перемещение блоков]) Вы можете копировать содержимое одной области памяти (=исходная область) в другую область памяти (=целевую область). Эта операция копирования не может прерываться другими действиями операционной системы.

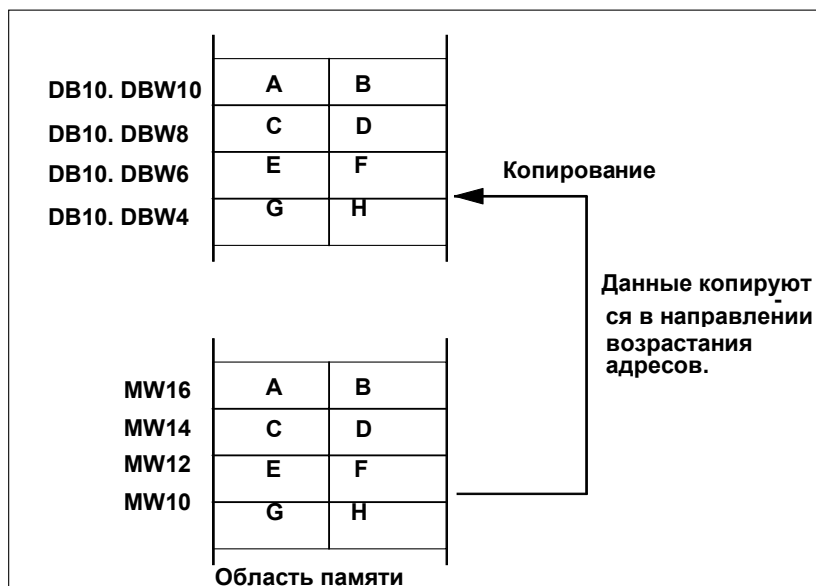
Используя SFC81 "UBLKMOV", Вы можете копировать все области памяти, кроме:

- следующих типов блоков: FB, SFB, FC, SFC, OB, SDB
- счетчиков
- таймеров
- областей памяти периферийных входов/выходов

Исходные параметры могут также быть включены в загружаемую область памяти несвязанных блоков данных (скомпилированных с опцией UNLINKED)!

### Примечание

Если в CPU есть SFC83, Вы должны использовать эту функцию для чтения блоков данных в загружаемой памяти, несвязанных с выполняемой задачей. Если Вы будете использовать для этой цели SFC20, то появится сообщение об ошибке W#16#8092.



**Возможность прерывания, время реакции на прерывание**

Копирование не может прерываться. Помните, что, если Вы используете SFC81 "UBLKMOV", то это может увеличить время реакции Вашего CPU на прерывание.

Параметр	Описание	Тип	Область памяти	Характеристика
SRCBLK	INPUT	ANY	I, Q, M, D, L	Определяет копируемую область памяти (исходную область). Массивы типа STRING недопустимы.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.
DSTBLK	OUTPUT	ANY	I, Q, M, D, L	Определяет область памяти, в которую будут копироваться данные (целевая область). Массивы типа STRING недопустимы.

**Примечание**

Исходная и целевая области не должны перекрываться. Если заданная целевая область больше, чем исходная область, то функция копирует в целевую область только такое количество данных, какое содержится в исходной области.

Если указанная целевая область меньше, чем исходная область, то функция копирует только такое количество данных, какое может быть записано в целевой области.

Если указатель ANY (источника или адресата) имеет тип BOOL, то заданная длина должна делиться на 8; иначе SFC не будет выполняться.

Если указатель ANY имеет тип STRING, то заданная длина должна быть равна 1.

**Информация об ошибках**

Код ошибки (W#16#...)	Характеристика
0000	Ошибки нет
8091	Исходная область находится в несвязанном блоке данных
8092	Операция "Copy Variable" (копирование переменной) не может быть выполнена, так как требует доступа к несвязанному блоку данных. Здесь необходимо использовать функцию SFC83.

### 3.3 Инициализация области памяти с помощью SFC21 "FILL"

#### Описание

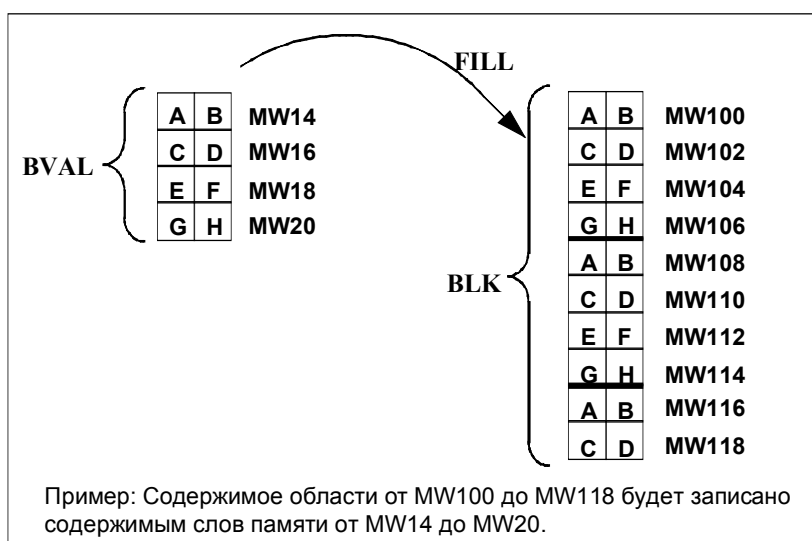
С помощью SFC 21 "FILL" Вы можете предварительно заполнить область памяти (целевую область) содержимым другой области памяти (исходной области). SFC копирует содержимое в заданную целевую область, пока эта область памяти не будет полностью записана.

#### Примечание

Исходный и целевой массив не должны перекрываться.

Если целевая область, подлежащая инициализации, не является целым кратным длины входного параметра BVAL, то она, тем не менее, записывается до последнего байта.

Если целевая область, подлежащая инициализации, меньше, чем исходная область, то копируется лишь столько данных, сколько может быть записано в целевую область.



#### Исключения

С помощью SFC 21 нельзя записывать значения в:

- следующие типы блоков: FB, SFB, FC, SFC, SDB,
- счетчики,
- таймеры,
- области памяти периферийных входов/выходов.

Параметр	Описание	Тип данных	Область памяти	Характеристика
BVAL	INPUT	ANY	I, Q, M, D, L	Параметр BVAL содержит значение или описание области, содержимое которой будет использовано для инициализации целевой области(исходная область). Массивы, относящиеся к типу данных STRING, недопустимы.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции появляется ошибка, то возвращаемое значение содержит код ошибки.
BLK	OUTPUT	ANY	I, Q, M, D, L	Параметр BLK содержит описание области, подлежащей инициализации (целевой области). Массивы, относящиеся к типу данных STRING, недопустимы.

### Входной параметр является структурой

Если в качестве входного параметра передается структура, то помните следующую особенность: STEP 7 всегда определяет длину структуры как четное число байтов. В результате структуре потребуется один байт дополнительного пространства в памяти, если Вы опишете структуру с нечетным числом байтов.

#### Пример

Структура описана следующим образом:

```
TYP_5_BYTE_STRUCTURE : STRUCT
    BYTE_1_2 : WORD
    BYTE_3_4 : WORD
    BYTE_5 : BYTE
END_STRUCT
```

Описанная структура "TYP\_5\_BYTE\_STRUCTURE" требует 6 байтов памяти.

### 3.4 Создание блока данных с помощью SFC22 "CREAT\_DB"

#### Описание

С помощью SFC22 "CREAT\_DB" (create data block [создать блок данных]), Вы создаете блок данных, который не содержит предварительно установленных значений. Этот SFC создает блок данных выбираемой длины с номером, взятым из заданного диапазона. SFC присваивает DB минимально возможный номер из заданного диапазона. Если Вы хотите создать DB с конкретным номером, то просто выберите диапазон, указывая одно и то же значение для верхней и нижней границы. Вы не можете назначить номер, если DB с таким же номером уже существует в программе пользователя. Длина DB должна быть четным числом байтов.

#### Возможность прерывания

SFC22 "CREAT\_DB" может прерываться более приоритетными ОВ. Если SFC22 "CREAT\_DB" вновь вызывается в ОВ более высокого приоритета, то этот вызов отвергается с кодом ошибки W#16#8091.

Параметр	Описание	Тип данных	Область памяти	Характеристика
LOW_LIMIT	INPUT	WORD	I, Q, M, D, L, константа	Нижнее граничное значение является минимальным номером в диапазоне номеров, которые Вы можете присвоить своему блоку данных.
UP_LIMIT	INPUT	WORD	I, Q, M, D, L, константа	Верхнее граничное значение является максимальным номером в диапазоне номеров, которые Вы можете присвоить своему блоку данных.
COUNT	INPUT	WORD	I, Q, M, D, L, константа	Числовое значение задает количество байтов данных, которое Вы хотите зарезервировать для своего блока данных. Здесь Вы должны задать четное число байтов (максимум 65534).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции появляется ошибка, то возвращаемое значение содержит код ошибки.
DB_NUMBER	OUTPUT	WORD	I, Q, M, D, L	Номер блока данных является номером созданного блока данных. Если происходит ошибка (бит 15 в RET_VAL был установлен), то в DB_NUMBER вводится значение 0.

## Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибки.
8091	Вы выполнили вложенный вызов SFC22.
8092	<p>Функция "Создать DB" не может быть в данный момент выполнена, так как:</p> <ul style="list-style-type: none"> <li>• в данный момент активна функция "Сжатие памяти пользователя"</li> <li>• в данный момент активна функция "Сохранение программы пользователя"</li> <li>• основной CPU выполняет функции соединения или обновления.</li> <li>• программа из ПО WinAC Software CPU обнаружила ошибку в ОС компьютера, в котором инсталлирована система WinAC.</li> </ul>
80A1	<p>Ошибка в номере DB:</p> <ul style="list-style-type: none"> <li>• Номер равен 0.</li> <li>• Номер превышает количество DB для конкретного CPU.</li> <li>• Параметр lower limit [нижняя граница] &gt; upper limit [верхняя граница].</li> </ul>
80A2	<p>Ошибка в длине DB:</p> <ul style="list-style-type: none"> <li>• Длина равна 0.</li> <li>• Длина была задана как нечетное число.</li> <li>• Длина больше, чем допускает CPU.</li> </ul>
80B1	Нет свободных номеров DB.
80B2	Недостаточно свободной памяти.
80B3	Недостаточно непрерывной памяти (устранение: провести сжатие!)



### 3.5 Удаление блока данных с помощью SFC23 "DEL\_DB"

#### Описание

С помощью SFC 23 "DEL\_DB" (delete data block) Вы удаляете блок данных, находящийся в рабочей памяти, и, если имеется, в загрузочной памяти CPU. Удаляемый DB не должен быть открыт ни в текущем, ни в каком-либо более низком классе приоритета, иными словами, он не должен быть внесен ни в один из двух регистров DB, ни в В-стек. В противном случае CPU при вызове SFC 23 переходит в состояние STOP.

Следующая таблица поясняет, когда DB может быть удален с помощью SFC23 "DEL-DB".

Если...	то...
DB был создан путем вызова SFC22 "CREAT_DB",	SFC23 может его удалить.
DB был передан в CPU посредством STEP 7 и не был создан с ключевым словом UNLINKED,	SFC23 может его удалить.
DB находится на флэш-карте,	SFC23 не может его удалить.

#### Возможность прерывания

SFC23 "DEL\_DB" может быть прерван более высокими классами приоритета. Если этот SFC вызывается там вновь, то этот второй вызов прерывается и в RET\_VAL записывается W#16#8091.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DB_NUMBER	INPUT	WORD	I, Q, M, D, L, константа	Номер удаляемого DB
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке

#### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8091	Вызовы SFC23 были вложенными, и была превышена максимальная глубина вложения для используемого CPU.
8092	Функция "Стирание DB" в данный момент не может быть выполнена, так как <ul style="list-style-type: none"> <li>• в данный момент активна функция "Сжатие памяти пользователя"</li> <li>• в данный момент активна функция "Сохранение программы пользователя".</li> <li>• в данный момент активна функция "Выгрузка (upload)" в отношении DB, который необходимо уничтожить</li> <li>• основной CPU выполняет функции соединения или обновления.</li> <li>• программа из ПО WinAC Software CPU обнаружила ошибку в ОС компьютера, в котором инсталлирована система WinAC.</li> </ul>

80A1	Ошибка во входном параметре DB_NUMBER: выбранный фактический параметр <ul style="list-style-type: none"><li>• имеет значение 0</li><li>• больше, чем максимально допустимый номер DB для используемого CPU.</li></ul>
80B1	DB с заданным номером в CPU не существует.
80B2	DB с заданным номером был создан с ключевым словом UNLINKED.
80B3	DB находится на флэш-карте.
80B4	DB не может быть удален, так как является блоком F-библиотеки.
80C1	Функция уничтожения DB "Delete DB" не может быть выполнена в настоящее время из-за временного критического состояния параметров.

### 3.6 Тестирование блока данных с помощью SFC24 "TEST\_DB"

#### Описание

С помощью SFC24 "TEST\_DB" (test data block [тестировать блок данных]) Вы получаете информацию о блоке данных, находящемся в рабочей памяти CPU. SFC запрашивает сведения о количестве байтов данных в выбранном DB и проверяет, защищен ли DB от записи.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DB_NUMBER	INPUT	WORD	I, Q, M, D, L, константа	Номер тестируемого DB
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках
DB_LENGTH	OUTPUT	WORD	I, Q, M, D, L	Количество байтов данных, которое содержит выбранный DB.
WRITE_PROT	OUTPUT	BOOL	I, Q, M, D, L	Информация о признаке защиты выбранного DB от записи (1 означает, что он защищен от записи).

#### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
80A1	Ошибка во входном параметре DB_NUMBER: выбранный фактический параметр <ul style="list-style-type: none"> <li>• имеет значение 0</li> <li>• больше, чем максимально допустимый номер DB для используемого CPU.</li> </ul>
80B1	DB с заданным номером в CPU не существует.
80B2	DB был создан с ключевым словом UNLINKED.

### 3.7 Сжатие памяти пользователя с помощью SFC25 "COMPRESS"

#### Пропуски в памяти

Пропуски могут возникать как в рабочей, так и в загрузочной памяти вследствие многократного удаления и повторной загрузки блоков. Эти пропуски сокращают эффективную область памяти.

#### Описание

С помощью SFC 25 "COMPRESS" Вы запускаете сжатие раздела ОЗУ как рабочей, так и загрузочной памяти. Функция сжатия здесь та же самая, что и при внешнем запуске в режиме RUN-P (положение переключателя режимов работы).

Если сжатие было запущено извне и все еще активно, то вызов SFC25 приведет к появлению сообщения об ошибке.

---

#### Примечание

Блоки данных, имеющие длину более 1000 байтов, не сдвигаются функцией SFC25 "COMPRESS". Это значит, что после сжатия в рабочей памяти могут остаться пропуски.

---

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке
BUSY	OUTPUT	BOOL	I, Q, M, D, L	Показывает, активна ли еще функция сжатия, запущенная вызовом SFC25. (1 означает, что активна)
DONE	OUTPUT	BOOL	I, Q, M, D, L	Показывает, успешно ли завершился процесс сжатия, который был запущен посредством SFC 25 (1 означает "успешно").

### Проверка функции сжатия

При однократном вызове SFC 25 "COMPRESS" запускается функция сжатия; однако Вы не можете проверить, успешно ли выполнилось сжатие.

Если Вы хотите проверить функцию сжатия, выполните описанные ниже шаги:

Вызовите SFC25 циклически. После каждого вызова сначала оценивайте параметр RET\_VAL. Если его значение равно 0, можно оценить параметры BUSY и DONE. Если BUSY = 1, а DONE = 0, то это указывает, что функция сжатия еще активна. Когда BUSY примет значение 0, а DONE - значение 1, то это указывает, что функция сжатия успешно завершена. Если потом вновь вызвать SFC 25, то вновь запустится сжатие.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было. Функция сжатия была запущена SFC 25. Только в этом случае имеет смысл оценивание в программе пользователя выходных параметров BUSY и DONE (см. выше).
8091	Функция сжатия была запущена извне и еще активна.
8092	Функция "Сжатие памяти пользователя" в данный момент не может быть выполнена, так как <ul style="list-style-type: none"> <li>• была запущена извне и все еще активна функция "Удаление блока данных"</li> <li>• функция тестирования и запуска в настоящее время требует конкретного блока (например, статус)</li> <li>• была запущена извне и все еще активна функция "Копирование блоков".</li> <li>• основной CPU выполняет функции соединения или обновления.</li> </ul>

### 3.8 Передача заменяющего значения в аккумулятор 1 с помощью SFC44 "REPL\_VAL"

#### Описание

С помощью SFC44 "REPL\_VAL" (replace value [заменить значение]) Вы передаете значение в аккумулятор 1 класса приоритета, вызвавшего ошибку.

#### Ограничение: только в ОБ синхронных ошибок

Вы можете вызывать SFC 44 "REPL\_VAL" только в ОБ синхронных ошибок (OB121, OB122).

#### Пример применения

Если модуль ввода поврежден до такой степени, что из него более нельзя считывать значения, то при каждом обращении к этому модулю запускается OB122. В OB122 с помощью SFC44 "REPL\_VAL" в аккумулятор 1 прерванного класса приоритета может быть передано подходящее значение, так что программа может продолжать работу с этим заменяющим значением. Информация, нужная для выбора заменяющего значения (например, блок, в котором встретилась ошибка, или поврежденный операнд), находится локальных переменных OB122.

Параметр	Описание	Тип данных	Область памяти	Характеристика
VAL	INPUT	DWORD	I, Q, M, D, L, константа	Заменяющее значение
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции встретилась ошибка, то возвращаемое значение содержит код ошибки.

#### Информация об ошибках

Код ошибки (W#16#....)	Описание
0000	Ошибок не было. Было введено заменяющее значение.
8080	SFC44 не вызывался ни одним из ОБ синхронных ошибок (OB121, OB122).

### 3.9 Генерация блоков данных в загружаемой памяти с помощью SFC82 "CREA\_DBL"

#### Описание

С помощью SFC82 "CREA\_DBL" (create date block in load memory [создание DB в загружаемой памяти]) Вы можете создать новый блок данных в загружаемой памяти (Memory Card – карта памяти). SFC82 генерирует DB с размером, заданным по умолчанию, и с номером из специального ряда. Функция SFC82 назначает наименьший из возможных номеров блоку. Вы можете задать номер DB путем назначения данного номера верхнему и нижнему граничным значениям для вышеуказанного специального ряда номеров. Невозможно задать уже используемый номер DB. Если уже существует блок с данным номером в рабочей памяти и/или загружаемой памяти, то будет выдано сообщение об ошибке.

---

#### Примечание

С помощью SFC24 "TEST\_DB" Вы можете определить, существует ли уже DB с заданным номером.

---

Содержание области данных, на которую указывает параметр SRCBLK записывается в DB. Эта область данных должна быть DB с типом данных BLOCK\_DB или областью из DB. Чтобы обеспечить корректность, Вы не должны изменить(заменить) эту область данных, во время работы SFC 82. С помощью SFC 82 может быть создан и инициализирован DB с атрибутом READ\_ONLY.

SFC 82 не изменяет контрольную сумму программы пользователя.

#### Принцип работы

SFC82 "CREA\_DBL" работает асинхронно, т.е. время ее работы может перекрывать несколько вызовов функции. Запуск функции SFC82 осуществляется с REQ =1.

Состояние работы функции можно контролировать с помощью выходных параметров RET\_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET\_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: Запрос на генерацию блока данных
LOW_LIMIT	INPUT	WORD	I, Q, M, D, L	Наименьший номер для последовательности номеров из ряда для назначения вновь создаваемым DB
UP_LIMIT	INPUT	WORD	I, Q, M, D, L	Наибольший номер для последовательности номеров из ряда для назначения вновь создаваемым DB
COUNT	INPUT	WORD	I, Q, M, D, L	Значение счетчика определяет количество байтов данных, которые резервируются для DB. Здесь необходимо определить четное число байтов.
ATTRIB	INPUT	BYTE	I, Q, M, D, L	Характеристики DB:
				Бит 0 = 1: UNLINKED: (несвязанный): DB существует только в загружаемой памяти.
				Бит 1 = 1: READ_ONLY: (только чтение): DB защищен от записи.
				Бит 2 = 1: NON_RETAIN: (ременантность): DB нереманентный.
				Бит 3 ... 7: Резерв
SRCBLK	INPUT	ANY	D	Указатель на область данных, в которой DB инициализирован
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Код ошибки
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: (состояние работы): Процесс не завершен
DB_NUM	OUTPUT	WORD	I, Q, M, D, L	Номер сгенерированного DB

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибок не было
0081	Область назначения (целевая) больше области исходной (источника). Исходная информация полностью записывается в область назначения. Свободные байты области назначения заполняются нулями.
7000	Первый вызов с REQ=0: передача данных не активирована; параметр BUSY имеет значение 0.
7001	Первый вызов с REQ=1: передача данных не активирована; параметр BUSY имеет значение 1.
7002	Промежуточное обращение (REQ незначущий): передача данных уже активирована; BUSY имеет значение 1.



8081	Область назначения (целевая) меньше области исходной (источника). Область назначения полностью заполняется. Непоместившаяся в ней часть исходной информации игнорируется.
8091	Имеются вложенные вызовы функции SFC82
8092	Генерация DB не выполняется, т.к.: <ul style="list-style-type: none"> <li>• активирован процесс сжатия памяти приложения</li> <li>• Н CPU выполняет функции соединения или обновления</li> <li>• крах системы Windows NT (синий экран)</li> </ul>
8093	Для параметра SCRBLK нет блока или нет соответствующего выполняемого блока (параметр инициализации блока)
8094	Еще не поддерживается атрибут, который был определен для параметра ATTRIB
80A1	Ошибка нумерации DB: <ul style="list-style-type: none"> <li>• номер блока равен 0</li> <li>• минимальный номер для диапазона номеров больше максимального</li> </ul>
80A2	Ошибка размера DB: <ul style="list-style-type: none"> <li>• длина блока равна 0</li> <li>• длина блока равна нечетному числу</li> <li>• размер блока больше максимально допустимого для CPU</li> </ul>
80B1	Нет свободных номеров для DB
80B2	Недостаточно свободной памяти
80B3	Недостаточно непрерывной памяти (необходимо провести сжатие)
80BV	Недостаточно загружаемой памяти
80C0	Область назначения используется в настоящее время другой функцией SFC или функцией связи
80C3	В настоящее время требуемые ресурсы недоступны
8хуу	Общая ошибка программы, например: <ul style="list-style-type: none"> <li>• исходный DB не существует или доступен только в виде копии</li> <li>• исходная область в DB не существует</li> </ul>

### 3.10 Считывание информации из блока данных в загружаемой памяти с помощью SFC83 "READ\_DBL"

#### Описание

С помощью SFC83 "READ\_DBL" (read date block in load memory [чтение DB в загружаемой памяти]) Вы можете считывать данные из DB типа BLOCK\_DB или из DB в загружаемой памяти (Memory Card – карта памяти) в область назначения DB (DB назначения). DB назначения должен быть выполняемым (иметь атрибут UNLINKED = 0). Исходная область, из которой производится считывание не обязательно должна быть связанной (может иметь атрибут UNLINKED = 1). Содержимое загружаемой памяти не изменяется в процессе выполнения операции чтения.

Для корректной работы функции нельзя изменять область назначения во время работы SFC83.

На параметры SRCBLK и DSTBLK накладываются следующие ограничения:

- Для указателя ANY типа BOOL длина должна делиться на 8.
- Если указатель ANY имеет тип STRING, то заданная длина должна быть равна 1.

При необходимости Вы можете определить размер исходной области с помощью функции SFC24 "TEST\_DB".

---

#### Примечание

Функция SFC83 может работать асинхронно, поэтому следует избегать частого чтения переменных из загружаемой памяти.

---

#### Примечание

Будучи запущенной, функция SFC83 всегда должна без прерывания завершить работу, даже если используемые ею ресурсы запрашиваются заданием с более высоким приоритетом.

---

#### Принцип работы

Функция SFC83 предназначена для асинхронной работы, т.е. время ее работы может перекрывать несколько вызовов функции. Запуск функции SFC83 осуществляется с REQ =1.

Состояние работы функции можно контролировать с помощью выходных параметров RET\_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET\_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: Запрос на чтение блока данных
SRCBLK	INPUT	ANY	D	Указатель на область данных DB в загружаемой области, из которой производится чтение
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Код ошибки
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: (состояние работы): Процесс не завершен
DSTBLK	OUTPUT	ANY	D	Указатель на область данных целевого DB

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибок не было
8081	Область назначения (целевая) меньше области исходной (источника). Область назначения полностью заполняется. Непоместившаяся в ней часть исходной информации игнорируется.
7000	Первый вызов с REQ=0: передача данных не активирована; параметр BUSY имеет значение 0.
7001	Первый вызов с REQ=1: передача данных не активирована; параметр BUSY имеет значение 1.
7002	Промежуточное обращение (REQ незначущий): передача данных уже активирована; BUSY имеет значение 1.
0081	Область назначения (целевая) больше области исходной (источника). Исходная информация полностью записывается в область назначения. Значения свободных байтов области назначения не изменяются.
80C0	Область назначения используется в настоящее время другой функцией SFC или функцией связи
8093	Для параметра SRCBLK нет блока или нет соответствующего выполняемого блока (параметр инициализации блока)
80B1	Для параметра DSTBLK нет блока или нет соответствующего выполняемого блока (параметр инициализации блока)
80B4	DB с F-атрибутом не может быть изменен
80C3	В настоящее время требуемые ресурсы недоступны
8хуу	Общая ошибка программы

### 3.11 Запись блока данных в загружаемой памяти с помощью SFC84 "WRITE\_DBL"

#### Описание

С помощью SFC84 "WRITE\_DBL" (write date block in load memory [запись DB в загружаемой памяти]) Вы можете записать данные исходного DB в DB типа BLOCK\_DB или в область DB в загружаемой памяти (Memory Card – карта памяти). DB в загружаемой памяти (с параметром DSTBLK) может быть выполняемым или невыполняемым. Исходная область, из которой производится считывание, может быть типа BLOCK\_DB или DB-содержимым в рабочей памяти, следовательно, исходный DB (с параметром SRCBLK) должен быть связанным (выполняемым), т.е. с параметром UNLINKED = 0. Кроме того, исходный DB может быть сгенерирован с помощью функции SCF22 "CREAT\_DB".

Для корректной работы функции нельзя изменять область назначения во время работы SFC84.

На параметры SRCBLK и DSTBLK накладываются следующие ограничения:

- Для указателя ANY типа BOOL длина должна делиться на 8.
- Если указатель ANY имеет тип STRING, то заданная длина должна быть равна 1.

При необходимости Вы можете определить размер исходной области с помощью функции SFC24 "TEST\_DB".

Во время записи загруженного DB при первом входе в него изменяется контрольная сумма пользовательской программы

---

#### Примечание

Функция SFC84 может работать асинхронно, поэтому следует избегать частой записи переменных в загружаемой памяти. Частое использование данной функции, кроме того, укорачивает срок службы загружаемой памяти.

---

#### Принцип работы

SFC84 "WRITE\_DBL" предназначена для асинхронной работы, т.е. время ее работы может перекрывать несколько вызовов функции. Запуск функции SFC83 осуществляется с REQ =1.

Состояние работы функции можно контролировать с помощью выходных параметров RET\_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET\_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: Запрос на запись
SRCBLK	INPUT	ANY	D	Указатель на область данных DB, из которой производится чтение
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Код ошибки
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: (состояние работы): Процесс не завершен
DSTBLK	OUTPUT	ANY	D	Указатель на область данных целевого DB в загружаемой памяти

### Информация об ошибках

Код ошибки (W#16#....)	Описание
0000	Ошибок не было
8081	Область назначения (целевая) меньше области исходной (источника). Область назначения полностью заполняется. Непоместившаяся в ней часть исходной информации игнорируется.
7000	Первый вызов с REQ=0: передача данных не активирована; параметр BUSY имеет значение 0.
7001	Первый вызов с REQ=1: передача данных не активирована; параметр BUSY имеет значение 1.
7002	Промежуточное обращение (REQ незначущий): передача данных уже активирована; BUSY имеет значение 1.
0081	Область назначения (целевая) больше области исходной (источника). Исходная информация полностью записывается в область назначения. Значения свободных байтов области назначения не изменяются.
80C0	Область назначения используется в настоящее время другой функцией SFC или функцией связи
8092	Краш ОС Windows NT (синий экран)
8093	Для параметра SCRBLK нет блока или нет соответствующего выполняемого блока (параметр инициализации блока)
80B1	Для параметра DSTBLK нет блока или нет соответствующего выполняемого блока (параметр инициализации блока)
80B4	DB с F-атрибутом не может быть изменен
80C3	В настоящее время требуемые ресурсы недоступны
8хуу	Общая ошибка программы

## 3.12 Создание блока данных с помощью SFC85 "CREA\_DB"

### Описание

С помощью SFC85 "CREA\_DB" (создание блока данных) Вы можете создавать блок данных без значений по умолчанию в пользовательской программе. Вместо них DB будет содержать случайные данные. SFC создает блок данных с номером из определенной области и с заданными размерами. SFC присваивает наименьший возможный номер из этой области DB. Для создания DB с определенными номерами, введите значения для нижнего и верхнего предела для области номеров по умолчанию. Вы не можете использовать номера DB, уже использованные в пользовательской программе. Длина DB должна быть определена четным числом.

В зависимости от выбора параметра ATTRIB созданный DB может иметь свойство RETAIN или NON\_RETAIN:

- RETAIN означает, что DB создается в ретанентной части рабочей области (work memory). Это значит, что текущие значения DB сохраняются после каждого перехода выключение/включение питания и после каждого перезапуска ("теплого" - "warm").
- NON\_RETAIN означает, что DB создается в неретанентной части рабочей памяти (work memory). Это значит, что текущие значения DB после каждого перехода выключение/включение питания и после каждого перезапуска ("теплого" - "warm") будут неопределенными.

Если эти части рабочей памяти (work memory), не различаются, то параметр ATTRIB игнорируются. Это значит, что значения DB сохраняются после каждого перехода выключение/включение питания и после каждого перезапуска ("теплого").

**Возможность прерывания функции**

Системная функция SFC85 "CREA\_DB" может быть прервана ОВ с более высоким приоритетом. Если SFC85 "CREA\_DB" вновь вызывается в этом ОВ с высоким приоритетом, то такой вызов будет игнорирован с кодом ошибки W#16#8091.

Параметр	Объявление	Тип данных	Область памяти	Описание
LOW_LIMIT	INPUT	WORD	I, Q, M, D, L константа	Нижнее предельное значение - это наименьший номер из диапазона номеров, который Вы можете назначить для Вашего блока данных
UP_LIMIT	INPUT	WORD	I, Q, M, D, L константа	Верхнее предельное значение - это наименьший номер из диапазона номеров, который Вы можете назначить для Вашего блока данных
COUNT	INPUT	WORD	I, Q, M, D, L константа	Это значение показывает число байтов данных, которое Вы хотите назначить для Вашего блока данных. Вы должны определить четное число байтов (максимум 65534).
ATTRIB	INPUT	BYTE	I, Q, M, D, L константа	Атрибуты DB: <ul style="list-style-type: none"> <li>• V#16#00: RETAIN (перманентные данные)</li> <li>• V#16#04: NON_RETAIN (неперманентные данные)</li> </ul>
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время выполнения произошла ошибка, то возвращаемое значение функции будет содержать код ошибки.
DB_NUMBER	OUTPUT	WORD	I, Q, M, D, L	Значение параметра - это номер созданного блока данных. В случае ошибки (бит 15 значения RET_VAL был установлен) в DB_NUMBER вводится значение 0.

## Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Нет ошибок
8091	Вы вызвали вложенную функцию SFC85.
8092	Функция SFC85 "CREA_DB" в настоящее время не доступна, так как: <ul style="list-style-type: none"> <li>• активирована функция сжатия пользовательской памяти "Compress User Memory"</li> <li>• обнаружена ошибка в ОС компьютера, на котором установлено ПО WinAC.</li> </ul>
8094	Некорректное значение параметра ATTRIB
80A1	Ошибка в номере блока DB: <ul style="list-style-type: none"> <li>• Номер равен 0</li> <li>• Номер превышает верхний предел, определенный CPU для DB</li> <li>• Нижний предел больше верхнего предела</li> </ul>
80A2	Ошибка размера блока DB: <ul style="list-style-type: none"> <li>• Размер блока равен 0</li> <li>• Размер блока задан нечетным числом</li> <li>• Размер блока больше допустимого для CPU значения</li> </ul>
80B1	Нет свободных номеров для DB.
80B2	Не достаточно свободной памяти.
80B3	Не достаточно непрерывной свободной памяти (запустите функцию "сжатия" памяти - compress).
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL



## **4 SFC для управления выполнением программы**

## 4.1 Повторный запуск контроля времени цикла с помощью SFC43 "RE\_TRIGR"

### Описание

С помощью SFC43 "RE\_TRIGR" (retrigger watchdog [перезапустить контрольный таймер]) Вы можете повторно запустить контроль времени цикла.

### Параметры

SFC43 "RE\_TRIGR" не имеет параметров.

### Информация об ошибках

SFC43 "RE\_TRIGR" не предоставляет информации об ошибках.

## 4.2 Перевод CPU в STOP с помощью SFC46 "STP"

### Описание

С помощью SFC 46 "STP" (stop) Вы переводите CPU в состояние STOP.

### Параметры

SFC46 "STP" не имеет параметров.

### Информация об ошибках

SFC46 "STP" не предоставляет информации об ошибках.

### 4.3 Задержка исполнения программы пользователя с помощью SFC47 "WAIT"

#### Описание

С помощью SFC 47 "WAIT" Вы программируете задержки или времена ожидания в своей пользовательской программе. Вы можете запрограммировать времена ожидания величиной до 32767 мкс. Наименьшее возможное время ожидания зависит от конкретного CPU и равно времени исполнения SFC 47.

#### Возможность прерывания

SFC47 "WAIT" может прерываться ОВ более высокого приоритета.

#### Примечание

**(только для S7-300, но не для CPU 318)**

Время ожидания, запрограммированное с помощью SFC 47, - это минимальное время. Оно продлевается за счет времени исполнения вложенных классов приоритета и загрузки системы.

Параметр	Описание	Тип данных	Область памяти	Характеристика
WT	INPUT	INT	I, Q, M, D, L, константа	Параметр WT содержит время ожидания в мкс.

#### Информация об ошибках

SFC47 "WAIT" не предоставляет информации об ошибках.

## 4.4 Запуск прерывания многопроцессорной обработки с помощью SFC35 "MP\_ALM"

### Описание

Вызов SFC 35 "MP\_ALM" при многопроцессорной обработке запускает прерывание многопроцессорной обработки. Это приводит к синхронизированному запуску OB 60 во всех соответствующих CPU. В однопроцессорном режиме и при работе с сегментированной стойкой OB 60 запускается только в тех CPU, которые вызвали SFC 35.

С помощью входного параметра JOB Вы можете отобразить причину прерывания многопроцессорной обработки. Этот идентификатор задания передается на все задействованные CPU и может использоваться вами в OB 60 (1234).

Вы можете вызывать SFC 35 "MP\_ALM" в любом месте своей программы. Однако поскольку вызов имеет смысл только в режиме RUN, то при вызове в режиме STARTUP прерывание многопроцессорной обработки подавляется. Об этом Вам сообщается через значение функции.

Параметр	Описание	Тип данных	Область памяти	Характеристика
JOB	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор задания: возможные значения: от 1 до 15
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при исполнении функции возникает ошибка, то возвращаемое значение содержит код ошибки.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8090	Входной параметр JOB содержит недопустимое значение.
80A0	Еще не завершено исполнение OB60 вслед за последним прерыванием многопроцессорной обработки на локальном или ином CPU.
80A1	Неправильный рабочий режим (STARTUP [запуск] вместо RUN).

## 4.5 Управление функцией CiR с помощью SFC104 "CiR"

### Описание

SFC104 "CiR" используется для переконфигурирования системы в режиме выполнения RUN:

- Вы можете полностью отменить функцию CiR. В этом случае загрузка измененной конфигурации из PG в CPU всегда недоступна. Такая блокировка сохраняется, пока Вы не отмените ее с помощью SFC104 "CiR".
- Вы можете задать верхний предел времени синхронизации CiR для выключения функции CiR по условию. В этом случае загрузка измененной конфигурации из PG в CPU доступна, только если CPU требуется меньше времени, чем установленное предельное значение, для проверки модифицированной конфигурации системы.
- Вы можете определить, возможно ли включение (enable) функции CiR. Если возможно такое включение или если возможно при определенном условии, то параметр A\_FT также возвращает фактический верхний предел для времени синхронизации CiR.

### Примечание

Значения на выходах замораживаются на период CiR-синхронизации, а входы в это же время не проверяются.

### Параметры

Параметр	Описание	Тип данных	Область памяти	Описание
MODE	INPUT	BYTE	I, Q, M, D, L, константа	ID задания (Job ID) Возможные значения: <ul style="list-style-type: none"> <li>• 0: Функция информации</li> <li>• 1: Функция CiR разрешена (верхний предел времени CiR-синхронизации задается значением по умолчанию)</li> <li>• 2: Функция CiR полностью запрещена</li> <li>• 3: Функция CiR запрещена с условием. Определите верхний предел времени CiR-синхронизации в FRZ_TIME.</li> </ul>

FRZ_TIME	INPUT	TIME	I, Q, M, D, L, константа	"Freeze time" ("время замораживания") Верхний предел времени CiR-синхронизации в мс Разрешенный диапазон значений: 200...2500 мс (по умолчанию: 1000 мс) Примечание: FRZ_TIME доступно только в режиме MODE=3.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит во время обработки функции, то возвращаемое значение содержит код ошибки. В режиме MODE=0, RET_VAL содержит информацию о том, разрешается ли активация CiR (enable).
A_FT	OUTPUT	TIME	I, Q, M, D, L	Действующее в текущий момент значение верхнего предельного значения времени CiR-синхронизации.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок нет. (Этот код возможен только для MODE=1 или MODE=2 или MODE=3)
0001	CiR функция разрешена (enabled). (Этот код возможен только для MODE=0.)
0002	CiR полностью заблокирован. (Этот код возможен только для MODE=0.)
0003	CiR заблокирован с условием. (Этот код возможен только для MODE=0.)
8001	CPU не готов для работы с CiR. Вы используете H CPU в H системе (автономно), или стандартный CPU в мульти компьютерном режиме.
8002	Некорректное значение в MODE
8003	Некорректное значение в FRZ_TIME
8хуу	Общая информация об ошибках: Проверка ошибок с выходным параметром RET_VAL

### Пример применения SFC104

SFC104 "CIR" может использоваться, например, чтобы заблокировать функцию CiR в периоды, когда от CPU требуется максимальная производительность.

Для этого выполните следующий сегмент программы CPU перед тем, как возрастет вычислительная нагрузка на CPU:

- Вызовите SFC104 с MODE = 2 (полная отмена CiR-функции)  
После окончания "интенсивного" периода выполните следующую инструкцию CPU:
- Вызовите SFC104 с MODE = 1 (восстановление CiR-функции) или MODE = 3 (блокировка CiR-функции по условию)





## **5 SFC для управления системными часами**

## 5.1 Установка времени с помощью SFC0 "SET\_CLK"

### Описание

С помощью SFC0 "SET\_CLK" (set system clock [установить системные часы]) Вы устанавливаете время суток и дату в часах CPU. Вызов SFC0 запускает часы. Затем часы ведут отсчет от установленных значений времени и даты.

Если часы являются ведущими, то CPU при вызове SFC 0 также начинает синхронизацию времени. Интервалы синхронизации устанавливаются с помощью STEP 7.

Параметр	Описание	Тип данных	Область памяти	Характеристика
PDT	INPUT	DT	D, L	На входе PDT вводятся дата и время суток, которые Вы хотите установить.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции появляется ошибка, то возвращаемое значение содержит код ошибки.

### Дата и время

Дата и время суток вводятся как тип данных DT. Пример: для установки 15-го января 1995 года, 10 час. 30 мин. и 30 секунд Вам следует ввести: DT#1995-01-15-10:30:30. Время может быть введено только с точностью до секунд. День недели рассчитывается функцией SFC0 "SET\_CLK" исходя из даты.

Помните, что тип данных DT вначале должен быть образован с помощью FC 3 "D\_TOD\_DT", прежде чем Вы сможете передать ему входные параметры.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибок
8080	Ошибка в дате
8081	Ошибка во времени

## 5.2 Считывание времени с помощью SFC1 "READ\_CLK"

### Описание

С помощью SFC 1 "READ\_CLK" (read system clock [читать системные часы]) считываются текущая дата и время с системных часов CPU.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при обработке функции появляется ошибка, то возвращаемое значение содержит код ошибки.
CDT	OUTPUT	DT	D,L	На выходе CDT выводятся текущая дата и текущее время суток.

### Информация об ошибках

Как оценить информацию об ошибках, содержащуюся в параметре RET\_VAL, объяснено в главе 1. Эта глава содержит также общую информацию об ошибках SFC. SFC1 не выводит специфическую информацию об ошибках.

### 5.3 Синхронизация ведомых часов с помощью SFC48 "SNC\_RTСВ"

#### Определение: Синхронизация ведомых часов

Под синхронизацией ведомых часов понимают передачу даты и времени суток от ведущих часов некоторого сегмента шины (например, К- шины S7-400, MPI, задней шины S7) на все ведомые часы этого сегмента шины.

#### Описание

С помощью SFC 48 "SNC\_RTСВ" (synchronize real time clocks [синхронизация часов реального времени]) Вы синхронизируете все имеющиеся на сегменте шины ведомые часы. Предпосылкой успешной синхронизации является вызов SFC 48 в CPU, часы реального времени которого были назначены ведущими часами по крайней мере для одного сегмента шины. Соответствующие параметры назначаются с помощью STEP 7.

Синхронизация ведомых часов со стороны системы (циклически по истечении выбранного интервала времени) происходит независимо от вызовов SFC 48.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при исполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.

#### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	При синхронизации ошибок не было.
0001	Имеющиеся часы не были назначены ведущими ни для одного сегмента шины.

## 5.4 Установка времени суток и TOD-статуса с помощью SFC100 "SET\_CLKS"

### Описание

С помощью SFC 100 "SET\_CLKS" осуществляется установка времени суток и TOD-статуса в CPU.

---

#### Внимание

Если системное время CPU не синхронизировано, используйте функцию SFC 100 "SET\_CLKS".

---

С помощью параметра MODE выбирается режим установки: будет ли изменено только время суток или только TOD-статус или оба параметра одновременно. Это показано в таблице ниже:

### Информация об ошибках

MODE (Режим) (W#16#...)	Объяснение
01	Корректировка времени суток. Вызов функции аналогичен вызову функции SFC 0 "SET_CLK". Входные параметры CORR, SUMMER и ANN 1 не проверяются.
02	Установка TOD-статуса. Входной параметр PDT не проверяется. Остальные входные параметры формируют следующие компоненты TOD-статуса: <ul style="list-style-type: none"> <li>• Величина коррекции, включая знак.</li> <li>• Ежечасный сигнал.</li> <li>• Индикатор летнего/зимнего времени</li> </ul> Точность установки времени (дискрет) определяется CPU. Бит ошибки синхронизации TOD-статуса получает значение FALSE (ЛОЖЬ). Время суток не изменяется.
03	Изменяются и время суток, и TOD-статус.

---

#### Внимание

Вы можете переопределить текущее значение TOD-статуса CPU (by retrieving SSL-ID W#16#0132 Index W#16#0008 with SFC 51 "RDSYSST").  
изменением SSL-ID W#16#0132 Index W#16#0008 с помощью функции SFC 51 "RDSYSST".

---

Параметр	Описание	Тип данных	Область памяти	Характеристика
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Возможные значения режима работы:
				<u>В#16#01:</u> Установка времени суток
				<u>В#16#02:</u> Установка TOD-статуса
				<u>В#16#03:</u> Установка времени суток и TOD-статуса
PDT	INPUT	DT	D, L	Время, принимаемое по умолчанию
CORR	INPUT	INT	I, Q, M, D, L, константа	Величина коррекции (по получасовой схеме). Возможные значения: -24 ... +26
SUMMER	INPUT	BOOL	I, Q, M, D, L, константа	Индикатор летнего/зимнего времени: <ul style="list-style-type: none"> <li>• 0 = Winter Time (зимнее время).</li> <li>• 1 = Summer Time (летнее время).</li> </ul>
ANN_1	INPUT	BOOL	I, Q, M, D, L, константа	Ежечасный сигнал.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при исполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8080	Значение MODE за пределами разрешенного диапазона.
8081	Значение CORR за пределами разрешенного диапазона (только для MODE = В#16#02 или MODE = В#16#03).
8082	Значение PDT за пределами разрешенного диапазона: некорректные значения даты и/или времени суток.

## **6 SFC для манипулирования счетчиками рабочего времени**

## 6.1 Счетчики рабочего времени

### Введение

CPU имеют в своем распоряжении несколько счетчиков рабочего времени (см. технические данные своего CPU).

- С помощью SFC 2, 3 и 4 Вы можете устанавливать и считывать значения счетчиков рабочего времени, а также их останавливать.
- Если CPU снабжен 32-разрядным счетчиком рабочего времени, то Вы можете устанавливать, останавливать и считывать его значения, используя функцию SFC101 "RTM".

---

### Примечание

Функции SFC 2, 3 и 4 также могут быть использованы с 32-разрядными счетчиками рабочего времени. Однако в этом случае счетчики могут работать только в режиме 16-разрядных счетчиков (с диапазоном значений: 0 ... 32767 часов).

См. также: запись данных подписка SSL ID W#16#0132 INDEX=W#16#000B.

---

### Применение

Вы можете использовать счетчик рабочего времени для самых разных приложений:

- для измерения продолжительности работы CPU
- для измерения продолжительности работы управляемой аппаратуры или подключенных устройств.

### Характеристики счетчика рабочего времени

При запуске счетчик рабочего времени начинает считать, начиная с последнего записанного значения. Если Вы хотите, чтобы он начал с другого начального значения, то Вы должны установить это значение с помощью SFC2 или SFC101 (в режиме MODE=4). Если CPU переходит в состояние STOP или Вы останавливаете счетчик рабочего времени, то CPU регистрирует текущее значение счетчика. Когда выполняется теплый или холодный рестарт CPU, нужно вновь запустить счетчик рабочего времени с помощью SFC3 или SFC101 (в режиме MODE=1).

### Диапазон значений

- для CPU с 16-разрядными счетчиками рабочего времени: 0...32 767 часов
- для CPU с 32-разрядными счетчиками рабочего времени: 0 ... (2E31) -1 = 2.147.483.647 часов



## 6.2 Управление счетчиками рабочего времени с помощью SFC101 "RTM"

### Описание

С помощью SFC101 "RTM" Вы можете устанавливать, останавливать и считывать значения 32-разрядного счетчика рабочего времени CPU.

Для выбора значений всех 32-разрядных счетчиков рабочего времени Вашего CPU вызовите SFC51 "RDSYSST" с SSL\_ID=W#16#0132 и INDEX=W#16#000B (для счетчиков рабочего времени 0 ... 7) или INDEX=W#16#000C (для счетчиков рабочего времени 8 ... 15). (См. также: запись данных подписка SSL ID W#16#0132 индекс W#16#000B.

Параметр	Описание	Тип данных	Область памяти	Характеристика
NR	INPUT	BYTE	I, Q, M, D, L, константа	Номер счетчика рабочего времени (возможные значения: 0 ... 15)
MODE	INPUT	BYTE	I, Q, M, D, L, константа	ID задания: <ul style="list-style-type: none"> <li>0: выбор (затем статус счетчика записывается в CQ, а текущее значение - в CV). После достижения значения (2E31) -1 часов, счетчик останавливается на верхнем значении, которое может быть отображено и выведено сообщение о переполнении ("Overflow").</li> <li>1: запуск (с последнего значения)</li> <li>2: стоп ("stop")</li> <li>4: установка (на значение в PV)</li> <li>5: установка (на значение в PV) с последующим запуском</li> <li>6: установка (на значение в PV) с последующим переходом в "stop"</li> </ul>
PV	INPUT	DINT	I, Q, M, D, L, константа	Новое значение для счетчика рабочего времени
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Возвращаемое значение будет содержать код ошибки, если она происходит, когда обрабатывается функция.
CQ	OUTPUT	BOOL	I, Q, M, D, L	Статус счетчика (1: выполнение /running/)
CV	OUTPUT	DINT	I, Q, M, D, L	Текущее значение счетчика рабочего времени

### Совместимость с программами, разработанными для CPU с 16-разрядными счетчиками рабочего времени

Функции SFC 2 "SET\_RTM", 3 "CTRL\_RTM" и 4 "READ\_RTM" также могут быть использованы с 32-разрядными счетчиками рабочего времени. Однако в этом случае счетчики могут работать только в режиме 16-разрядных счетчиков (с диапазоном значений: 0 ... 32767 часов).

Запись данных подписка SSL ID W#16#0132 индекс W#16#0008 отображает состояние 32-разрядных счетчиков рабочего времени 0 ... 7 в 16-разрядном режиме. Это означает, что Вы можете продолжать использовать программы, разработанные для CPU с 16-разрядными счетчиками рабочего времени, которые используют подписание SSL ID W#16#0132 с индексом W#16#0008.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибок
8080	Ошибка номера счетчика рабочего времени
8081	Отрицательное число в параметре PV
8082	Переполнение счетчика рабочего времени
8091	Некорректное значение в выходном параметре MODE.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

### 6.3 Установка счетчика рабочего времени с помощью SFC2 "SET\_RTM"

#### Описание

С помощью SFC 2 "SET\_RTM" (set run-time meter [установить счетчик рабочего времени]) счетчик рабочего времени CPU устанавливается на выбранное значение. Количество счетчиков рабочего времени, которое Вы можете установить, зависит от используемого вами CPU.

Параметр	Описание	Тип данных	Область памяти	Характеристика
NR	INPUT	BYTE	I, Q, M, D, L, константа	Вход NR содержит номер счетчика рабочего времени, который Вы хотите установить (возможные значения: от 0 до 7).
PV	INPUT	INT	I, Q, M, D, L, константа	Вход PV содержит установку для счетчика рабочего времени (по умолчанию).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при исполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.

#### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибок
8080	Неправильный номер счетчика рабочего времени
8081	Параметру PV было передано отрицательное значение.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 6.4 Запуск и останов счетчика рабочего времени с помощью SFC3 "CTRL\_RTM"

### Описание

С помощью SFC 3 "CTRL\_RTM" (control run-time meter [управление счетчиком рабочего времени]) Вы можете запускать или останавливать счетчик рабочего времени CPU.

Параметр	Описание	Тип данных	Область памяти	Характеристика
NR	INPUT	BYTE	I, Q, M, D, L, константа	Вход NR содержит номер счетчика рабочего времени, который Вы хотите запустить или остановить (возможные значения: от 0 до 7).
S	INPUT	BOOL	I, Q, M, D, L, константа	Вход S запускает или останавливает счетчик рабочего времени. Установите состояние сигнала на "0", когда Вы хотите остановить счетчик. Установите состояние сигнала на "1", когда Вы хотите запустить счетчик.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции происходит ошибка, то возвращаемое значение содержит код ошибки.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибок
8080	Неправильный номер счетчика рабочего времени
8хху	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 6.5 Считывание значения счетчика рабочего времени с помощью SFC4 "READ\_RTM"

### Описание

С помощью SFC 4 "READ\_RTM" (read run-time meter [считать значение счетчика рабочего времени]) Вы считываете значения счетчика рабочего времени. SFC 4 в качестве выходных данных выдает текущее время счета и состояние счетчика, т.е. "остановлен" или "считает".

Если проходит более 32767 часов, то счетчик останавливается на значении 32767 и выводит сообщение об ошибке "overflow [переполнение]".

Параметр	Описание	Тип данных	Область памяти	Характеристика
NR	INPUT	BYTE	I, Q, M, D, L, константа	Вход NR содержит номер счетчика рабочего времени, значение которого Вы хотите прочесть (возможные значения: от 0 до 7).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время обработки функции происходит ошибка, то возвращаемое значение содержит код ошибки.
CQ	OUTPUT	BOOL	I, Q, M, D, L	Выход CQ указывает, работает счетчик рабочего времени или остановлен. Состояние сигнала "0" указывает, что счетчик рабочего времени остановлен. Состояние сигнала "1" указывает, что счетчик рабочего времени работает.
CV	OUTPUT	INT	I, Q, M, D, L	Выход CV указывает текущее значение счетчика рабочего времени.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибки
8080	Неправильный номер счетчика рабочего времени
8081	Переполнение счетчика рабочего времени
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 6.6 Считывание системного времени с помощью SFC64 "TIME\_TCK"

### Описание

С помощью SFC 64 "TIME\_TCK" (time tick [такт времени]) Вы можете прочитать системное время CPU. Системное время - это "счетчик времени", который считает циклически от 0 до максимума в 2147483647 мс. В случае переполнения системного времени счет ведется опять от нуля. Разрешение и точность системного времени составляют 1 мс для S7-400 и CPU 318 и 10 мс для всех остальных CPU S7-300. Системное время находится под воздействием только рабочих режимов CPU.

### Применение

Вы можете использовать системное время, например, для того, чтобы измерять длительность процессов путем сравнения результатов двух вызовов SFC 64.

### Системное время и режимы работы

Режим работы	Системное время...
Startup (Запуск)	... обновляется постоянно
RUN	
STOP	... останавливается и сохраняет текущее значение
Горячий рестарт (нет в S7-300 и S7-400 H)	... продолжается от значения, сохраненного при переходе в STOP
Теплый рестарт	... стирается и начинается опять с "0"
Холодный рестарт	

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	TIME	I, Q, M, D, L	Параметр RET_VAL содержит считанное системное время в диапазоне от 0 до $2^{31}-1$ мс.

### Информация об ошибках

SFC64 "TIME\_TCK" не предоставляет информации об ошибках.

## **7 SFC для передачи записей данных**

## 7.1 Запись и чтение записей данных

### Принцип

Некоторые модули имеют предназначенную только для записи область системных данных, в которую Ваша программа может передавать записи данных. Эта область содержит записи данных с номерами от 0 до максимум 240. Не каждый модуль содержит все записи данных (см. следующую таблицу).

Другие модули имеют область системных данных, предназначенную только для чтения, в которой Ваша программа может читать записи данных. Эта область содержит записи данных с номерами от 0 до максимум 240. Не каждый модуль содержит все записи данных (см. следующую таблицу).

### Примечание

Есть модули, имеющие обе области системных данных. Это физически отдельные области, у которых общей является только их логическая структура.

### Область системных данных, предназначенная только для записи

Следующая таблица показывает структуру области системных данных, предназначенной только для записи. Эта таблица показывает также, какими длинными могут быть отдельные записи данных и с помощью каких SFC они могут записываться.

Номер записи данных	Содержимое	Размер	Ограничение	Может записываться с помощью SFC
0	Параметры	У S7-300: от 2 до 14 байтов	может записываться только контроллером S7-400	56 "WR_DPARM" 57 "PARM_MOD"
1	Параметры	У S7-300: от 2 до 14 байтов Записи данных 0 и 1 вместе имеют ровно 16 байтов.	-	55 "WR_PARM" 56 "WR_DPARM" 57 "PARM_MOD"
от 2 до 127	Данные пользователя	не более, чем по 240 байтов каждая	-	55 "WR_PARM" 56 "WR_DPARM" 57 "PARM_MOD" 58 "WR_REC"
от 128 до 240	Параметры	не более, чем по 240 байтов каждая	-	55 "WR_PARM" 56 "WR_DPARM" 57 "PARM_MOD" 58 "WR_REC"



**Область системных данных, предназначенных только для чтения**

Следующая таблица показывает структуру области системных данных, предназначенных только для чтения. Эта таблица показывает также, какими длинными могут быть отдельные записи данных и с помощью каких SFC они могут читаться.

Номер записи данных	Содержимое	Размер	Может читаться с помощью SFC
0	Специфические для модулей диагностические данные (устанавливаются, как правило, для всей системы)	4 байта	51 "RDSYSST" (SSL_ID 00B1H) 59 "RD_REC"
1	Специфические для канала диагностические данные (включая запись данных 0)	<ul style="list-style-type: none"> <li>• У S7-300: 16 байтов</li> <li>• У S7-400: от 7 до 220 байтов</li> </ul>	51 "RDSYSST" (SSL_ID 00B2H и 00B3H) 59 "RD_REC"
от 2 до 127	Данные пользователя	не более, чем по 240 байтов каждая	59 "RD_REC"
от 128 до 240	Диагностические данные	не более, чем по 240 байтов каждая	59 "RD_REC"

**Системные ресурсы**

Если Вы через короткие промежутки времени запускаете подряд несколько асинхронно протекающих процессов передачи записей данных, то распределение системных ресурсов операционной системой гарантирует, что все задания выполнятся и не будут влиять друг на друга.

Если используются все доступные системные ресурсы, то это отображается в RET\_VAL. Вы можете устранить эту нерегулярную ошибочную ситуацию простым повторением задания.

Максимальное количество "одновременно" активных заданий SFC зависит от CPU. За более подробной информацией обратитесь к [/70/](#) и [/101/](#).

## 7.2 Чтение определенных параметров с помощью SFC54 "RD\_DPARM"

### Описание

С помощью SFC54 "RD\_DPARM" (read defined parameter [читать определенный параметр]) Вы считываете запись данных адресуемого модуля с номером RECNUM из соответствующего SDB1ху. Считываемая запись данных вводится в целевую область, открытую с помощью параметра RECORD.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор (ID) адресной области: V#16#54 =Периферийный вход (PI) V#16#55 = Периферийный выход (PQ) Если модуль смешанный, то задайте ID области с наименьшим адресом. Если адреса одинаковые, то задайте V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический базовый адрес модуля. В случае смешанных модулей задайте меньший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных (разрешенные значения: от 0 до 240)
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция активна, происходит ошибка, то возвращаемое значение содержит код ошибки. А также: длина считываемой записи данных в байтах, если считываемая запись данных вмещается в область назначения, и при передаче не было ошибок.
RECORD	OUTPUT	ANY	I, Q, M, D, L	Область назначения для считываемой записи данных. Разрешается только тип данных BYTE.

### Информация об ошибках

См. раздел "Назначение параметров для модуля с помощью SFC57 "PARM\_MOD".

### 7.3 Чтение заранее определенных параметров с помощью SFC102 "RD\_DPARA"

#### Описание

С помощью SFC102 "RD\_DPARA" (read predefined parameters [считать предопределенные параметры]) Вы можете считывать запись данных выбранного модуля с номером RECNUM из соответствующего SDB1ху. Считываемая запись данных вводится в целевую область, открытую с помощью параметра RECORD.

#### Принцип работы

Функция SFC102 "RD\_DPARA" выполняется асинхронно, т.е. время ее работы может перекрывать несколько вызовов функции. Запуск функции SFC102 осуществляется с REQ =1.

Состояние работы функции можно контролировать с помощью выходных параметров RET\_VAL и BUSY.

Также для асинхронно работающих функций контролируйте значения REQ, RET\_VAL и BUSY.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: запрос на вызов функции
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический базовый адрес модуля. В случае смешанных модулей задайте меньший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных (разрешенные значения: от 0 до 240)
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция активна, происходит ошибка, то возвращаемое значение содержит код ошибки. А также: длина считываемой записи данных в байтах, если считываемая запись данных вмещается в область назначения, и при передаче не было ошибок.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: еще не завершено.
RECORD	OUTPUT	ANY	I, Q, M, D, L	Область назначения для считываемой записи данных. Разрешается только тип данных BYTE.

#### Информация об ошибках

См. раздел "Назначение параметров для модуля с помощью SFC57 "PARM\_MOD".

## 7.4 Запись динамических параметров с помощью SFC55 "WR\_PARM"

### Описание

С помощью SFC55 "WR\_PARM" (write parameter [записать параметр]) Вы передаете запись данных RECORD адресуемому модулю. Параметры, которые передаются модулю, **не** заменяют параметры этого модуля в соответствующем SDB, если они там существуют.

### Предпосылки

Передаваемая запись данных не должна быть статической:

- Это не должна быть запись данных 0 (запись данных 0 является статической во всей системе).
- Если происходит обращение к записи данных в SDB 100 – 129, то бит статики не должен быть установлен.

За дополнительной информацией о статических записях данных обратитесь к [/71/](#) и [/101/](#).

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: запрос на запись
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресной области: В#16#54 = периферийный вход (PI) В#16#55 = периферийный выход (PQ) Если модуль смешанный, то укажите идентификатор области самого младшего адреса. Если адреса одинаковы, укажите В#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический базовый адрес модуля. У смешанных модулей указывайте младший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных
RECORD	INPUT	ANY	I, Q, M, D, L	Запись данных
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при исполнении функции появляется ошибка, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Процесс записи еще не закончен.

### Входной параметр RECORD

Передаваемые данные читаются из параметра RECORD при первом вызове SFC. Если передача записи данных длится дольше, чем вызов, то содержимое параметра RECORD больше не является релевантным для последующих вызовов SFC (для того же самого задания).

### Информация об ошибках

См. раздел "Назначение параметров для модуля с помощью SFC57 "PARM\_MOD".

---

#### **Примечание (только для S7-400)**

Если происходит общая ошибка W#16#8544, то это показывает только, что был заблокирован доступ по крайней мере к одному байту области памяти входов/выходов, содержащей запись данных. Передача данных была продолжена

---

## 7.5 Запись параметров, установленных по умолчанию, с помощью SFC56 "WR\_DPARM"

### Описание

С помощью SFC56 "WR\_DPARM" (write default parameter [запись параметров, установленных по умолчанию]) Вы передаете запись данных с номером RECNUM из соответствующего SDB1ху адресуемому модулю. Для этой функции не имеет значения, является ли набор данных статическим или динамическим.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: запрос на запись
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресной области: V#16#54 = периферийный вход (PI) V#16#55 = периферийный выход (PQ) Если модуль смешанный, то укажите идентификатор области самого младшего адреса. Если адреса одинаковы, укажите V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический базовый адрес модуля. У смешанных модулей указывайте младший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при исполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: процесс записи еще не закончен.

### Информация об ошибках

См. раздел "Назначение параметров для модуля с помощью SFC57 "PARM\_MOD".

## 7.6 Назначение параметров для модуля с помощью SFC57 "PARM\_MOD"

### Описание

С помощью SFC57 "PARM\_MOD" (parametrize module [параметризовать модуль]) Вы передаете модулю все его записи данных, которые Вы сконфигурировали с помощью STEP 7 в соответствующем SDB. Для этой функции не имеет значения, являются ли эти записи данных статическими или динамическими.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: запрос на запись
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресной области: W#16#54 = периферийный вход (PI) W#16#55 = периферийный выход (PQ) Если модуль смешанный, то укажите идентификатор области самого младшего адреса. Если адреса одинаковы, укажите W#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический базовый адрес модуля. У смешанных модулей указывайте младший из двух адресов.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка возникает, когда функция активна, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: процесс записи еще не закончен.

### Информация об ошибках

"Реальная" информация об ошибках (коды ошибок W#16#8xyz) может быть разделена на два класса:

- Нерегулярные ошибки (коды ошибок: W#16#80A2 ... 80A4, 80Cх):  
Для этого типа ошибки возможно, что она будет устранена без Вашего вмешательства, иными словами, рекомендуется вызвать этот SFC вновь (если необходимо, более одного раза).  
Примером нерегулярной ошибки является ситуация, когда требуемые ресурсы в данный момент времени используются (W#16#80C3).
- Постоянные ошибки (коды ошибок: W#16#809х, 80A1, 80Bх, 80Dх):  
Этот тип ошибки не может быть устранен без Вашего вмешательства. Повторный вызов SFC будет успешным только после устранения ошибки. Примером постоянной ошибки является ввод неверной длины в параметр RECORD (W#16#80B1).

Специальная информация об ошибках для SFC54 "RD\_DPARM", SFC55 "WR\_PARM", SFC56 "WR\_DPARM" и SFC57 "PARM\_MOD".

Код ошибки (W#16#...)	Объяснение	Ограничение
7000	Первый вызов с REQ=0: нет активных процессов передачи данных; BUSY имеет значение 0.	-
7001	Первый вызов с REQ=1: запущен процесс передачи данных; BUSY имеет значение 1.	Децентрализованная периферия
7002	Промежуточный вызов (REQ не имеет значения): процесс передачи данных активен; BUSY имеет значение 1.	Децентрализованная периферия
8090	Заданный логический базовый адрес недействителен: Нет соответствия в SDB1/SDB2x или нет базового адреса.	-
8092	В ссылке ANY задан тип, отличный от BYTE.	Только в S7-400 для SFC54 "RD_PARM" и SFC55 "WR_PARM"
8093	Этот SFC недопустим для модуля, заданного в LADDR и IOID (разрешены следующие модули: модули S7-300 для S7-300, модули S7-400 для S7-400, модули S7-DP для S7-300 и S7-400).	-
80A1	Отрицательная квитанция при передаче записи данных модулю (во время передачи модуль был вытасчен или стал неисправным).	1)
80A2	Ошибка протокола DP на уровне 2, возможно, неисправность аппаратуры/интерфейса в slave-устройстве DP	Децентрализованная периферия <sup>1)</sup>
80A3	Ошибка протокола DP на уровне интерфейса пользователя/пользователь	Децентрализованная периферия <sup>1)</sup>
80A4	Ошибка связи на коммуникационной шине.	Ошибка происходит между CPU и внешним интерфейсным модулем DP <sup>1)</sup>
80B0	SFC неприменима для этого типа модуля, модуль не распознает эту запись данных.	1)
80B1	Неправильная длина передаваемой записи данных. У SFC54 "RD_PARM": длина целевой области, открытой параметром RECORD, слишком коротка.	-
80B2	Сконфигурированный слот не занят.	1)
80B3	Фактический тип модуля не совпадает с требуемым типом модуля в SDB1.	1)
80C1	Данные предшествующего задания на запись в модуле для той же самой записи данных еще не обработаны модулем.	1)
80C2	В данный момент времени модуль обрабатывает максимально возможное для CPU количество заданий.	1)
80C3	В данный момент времени требуемые ресурсы (память и т.д.) заняты.	1)



Код ошибки (W#16#...)	Объяснение	Ограничение
80C4	Внутренняя нерегулярная ошибка. Функция не может быть выполнена. <ul style="list-style-type: none"> <li>Повторите запуск задачи. При частом повторении ошибки проверьте Вашу систему на предмет электрических помех.</li> </ul>	1)
80C5	Децентрализованная периферия недоступна.	Децентрализованная периферия <sup>1)</sup>
80C6	Передача записи данных была остановлена из-за прерывания выполнения класса приоритета (горячий рестарт или фоновый режим).	Децентрализованная периферия <sup>1)</sup>
80D0	В соответствующем SDB нет записи для модуля.	-
80D1	Номер записи данных для модуля не сконфигурирован в соответствующем SDB. (Номера записей данных $\geq 241$ отклоняются STEP 7).	-
80D2	В соответствии с идентификатором типа модулю не могут быть назначены параметры.	-
80D3	SDB не может быть назначен, так как он не существует.	-
80D4	Ошибка структуры SDB: Внутренний указатель SDB указывает на значение за пределами SDB.	Только у S7-300
80D5	Запись данных является статической.	Только у SFC55 "WR_PARM"
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL	8хуу

1) Не бывает в SFC54 "RD\_DPARM"

## 7.7 Внесение записи данных с помощью SFC58 "WR\_REC"

### Описание

С помощью SFC58 "WR\_REC" (write record [внести запись]) Вы передаете запись данных, содержащуюся в RECORD, адресуемому модулю. Вы запускаете задание на запись, присваивая входному параметру REQ значение 1 при вызове SFC58. Если задание на запись смогло выполниться немедленно, то SFC возвращает в выходном параметре BUSY значение 0. Если BUSY имеет значение 1, то задание на запись еще не завершено.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: запрос на запись
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресной области: V#16#54 = периферийный вход (PI) V#16#55 = периферийный выход (PQ) Если модуль смешанный, то укажите идентификатор области самого младшего адреса. Если адреса одинаковы, укажите V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес модуля. У смешанных модулей указывайте младший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных (допустимые значения: от 2 до 240)
RECORD	INPUT	ANY	I, Q, M, D, L	Запись данных. Допустимым является только тип данных BYTE.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка появляется, когда функция активна, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: процесс записи еще не закончен.

### Входной параметр RECORD

Передаваемые данные читаются из параметра RECORD при первом вызове SFC. Если передача записи данных длится дольше, чем вызов, то содержимое параметра RECORD больше не является релевантным для последующих вызовов SFC (для того же самого задания).

### Информация об ошибках

См. раздел "Считывание записей данных с помощью функции SFC 59 "RD\_REC".

#### Примечание (только для S7-400)

Если происходит общая ошибка W#16#8544, то это показывает только то, что был заблокирован доступ по крайней мере к одному байту области памяти входов/выходов, содержащей запись данных. Передача данных была продолжена.

## 7.8 Чтение записи данных с помощью SFC59 "RD\_REC"

### Описание

С помощью SFC59 "RD\_REC" (read record [читать запись]) Вы считываете запись данных с номером RECNUM из адресуемого модуля. Задание на чтение запускают, вызывая SFC59 и присваивая входному параметру REQ значение 1. Если задание на чтение может быть выполнено немедленно, то SFC возвращает в выходном параметре BUSY значение 0. Если BUSY имеет значение 1, то задание на чтение еще не завершилось. Считываемая запись данных вводится в область назначения, указанную параметром RECORD, если передача данных была свободна от ошибок.

---

### Примечание

Если Вы считываете запись данных с номером выше 1 из FM или CP, купленного до февраля 1997 (называемого ниже "старым модулем"), то реакция SFC59 отличается от реакции в случае нового модуля. Эта особая ситуация описана в разделе "Использование старых FM и CP S7-300 с номерами записей данных выше 1".

---

---

### Примечание

Если ведомое устройство DPV1 сконфигурировано с использованием GSD-файла (GSD изд. 3), а DP-интерфейс ведущего DP-устройства установлен в режим "S7 compatible", тогда записи данных не должны считываться из входных/выходных модулей с помощью SFC59 в пользовательской программе. Причина этого в том, что в этом случае ведущее DP-устройство обращается к некорректному слоту (сконфигурированный слот + 3).

Способ решения проблемы: Установите DP-интерфейс ведущего DP-устройства установлен в режим "DPV1".

---

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: Запрос на чтение
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор (ID) адресной области: V#16#54 = Периферийный вход (PI) V#16#55 = Периферийный выход (PQ) Если модуль смешанный, то задайте ID области с наименьшим адресом. Если адреса одинаковые, то задайте V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес модуля. В случае смешанных модулей задайте меньший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных (разрешенные значения от 0 до 240)
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция активна, происходит ошибка, то возвращаемое значение содержит код ошибки. Если область назначения больше, чем переданная запись данных, и при передаче не было ошибок, то вводится также длина фактически переданной записи данных в байтах (возможные значения: от +1 до +240).
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Чтение еще не закончилось.
RECORD	OUTPUT	ANY	I, Q, M, D, L	Область назначения для считываемой записи данных. В случае асинхронного выполнения SFC59 убедитесь, что фактические параметры RECORD имеют одинаковую информацию о длине во всех вызовах. Разрешен только тип данных BYTE.

### Выходной параметр RET\_VAL

- Если во время выполнения функции произошла ошибка, то возвращаемое значение содержит код ошибки.
- Если ошибок не было, то RET\_VAL содержит:
  - 0: если вся область назначения была заполнена данными из выбранной записи данных (запись данных может быть также неполной).
  - длину фактически переданной записи данных в байтах (возможные значения: от +1 до +240), если область назначения больше, чем переданная запись данных.

---

**Примечание**

ошибка W#16#8545, то это указывает только на то, что был заблокирован доступ, по крайней мере, к одному байту области памяти входов/выходов, содержащей запись данных. Запись данных была правильно считана модулем и записана в область памяти входов/выходов.

---

**Установка RECORD**

---

**Примечание**

Если Вы хотите гарантировать, чтобы всегда читалась полная запись данных, то выберите область назначения длиной 241 байт. Если передача данных свободна от ошибок, то RET\_VAL содержит фактическую длину записи данных.

---

**Использование старых FM и CP S7-300 с номерами записей данных выше 1**

Если Вы хотите считывать с помощью SFC59 "RD\_REC" запись данных с номером выше 1 из старого FM S7-300 или старого CP S7-300, то помните о следующем:

- Если область назначения больше, чем фактическая длина требуемой записи данных, то никакие данные не вводятся в RECORD. RET\_VAL имеет значение W#16#80B1.
- Если область назначения меньше, чем фактическая длина требуемой записи данных, то CPU считывает столько байтов, начиная от начала записи, сколько определено в информации о длине RECORD, и вводит это количество байтов в RECORD. RET\_VAL имеет значение 0.
- Если длина, заданная в RECORD, равна фактической длине требуемой записи данных, то CPU считывает запись данных и вводит ее в RECORD. RET\_VAL имеет значение 0.

**Информация об ошибках**

"Реальные" данные об ошибках (коды ошибок W#16#8xyz) в следующей таблице могут быть разделены на два класса:

- Нерегулярные ошибки (коды ошибок от W#16#80A2 до 80A4, 80Cх):  
При этом типе ошибки возможно, что ошибка будет устранена без принятия вами каких-либо мер, другими словами, целесообразно вновь вызвать SFC (в случае необходимости более одного раза).  
Примером нерегулярной ошибки является случай, когда требуемые ресурсы в настоящее время используются (W#16#80C3).
- Постоянные ошибки (коды ошибки W#16#809х, 80A1, 80Bх, 80Dх):  
Ошибка этого типа не будет устранена без принятия вами мер. Повторный вызов SFC будет успешным только после того, как ошибка устранена.  
Примером постоянной ошибки является ввод неправильной длины в RECORD (W#16#80B1).

**Примечание**

Если Вы передаете записи данных ведомому устройству DPV1 с помощью SFC58 "WR\_REC" или, если Вы выбираете записи данных из ведомого устройства DPV1 подчиненный с SFC59 "RD\_REC", и если это устройство DPV1 функционирует в режиме DPV1, то ведущее устройство DP проверяет информацию об ошибках, которую оно получает от ведомого устройства DPV1, следующим образом:

Если код ошибки лежит в диапазонах W#16#8000 ... W#16#80FF или W#16#F000 ... W#16#FFFF, то ведущее устройство DP передает информацию об ошибках в SFC. Если код ошибки лежит вне указанных диапазонов, то CPU передает значение W#16#80A2 в SFC и приостанавливает ведомое устройство.

Для описания информации об ошибках, полученной от DPV1-устройства, см. раздел "Получение прерывания от ведомого устройства DP с помощью SFB 54 "RALRM" STATUS (состояние) [3].

Специальная информация об ошибках для SFC58 "WR\_REC" и SFC59 "RD\_REC".

Код ошибки (W#16#...)	Объяснение	Ограничение
7000	Первый вызов с REQ=0: Передача данных не активна; BUSY имеет значение 0.	-
7001	Первый вызов с REQ=1: Передача данных не активна; BUSY имеет значение 1.	Децентрализованная периферия
7002	Промежуточный вызов (REQ не имеет значения): Передача данных уже активна; BUSY имеет значение 1.	Децентрализованная периферия
8090	Задан недействительный логический базовый адрес: Нет назначения в SDB1/SDB2X или нет базового адреса.	-
8092	Тип, указанный в ссылке ANY, не BYTE.	Только S7-400
8093	Этот SFC не разрешен для модуля, определяемого LADDR и IOID (разрешены следующие модули: модули S7-300 для S7-300, модули S7-400 для S7-400, модули S7-DP для S7-300 и S7-400).	-
80A0	Отрицательное подтверждение при считывании из модуля <ul style="list-style-type: none"> <li>модуль был снят во время задания на чтение или неисправен;</li> <li>для H-систем: односторонние I/O резервного CPU недоступны (напр., резервный CPU в режиме STOP).</li> </ul>	Только в случае SFC59 "RD_REC"

Код ошибки (W#16#...)	Объяснение	Ограничение
80A1	Отрицательное подтверждение при передаче записи данных в модуль <ul style="list-style-type: none"> <li>модуль был снят во время задания на чтение или неисправен;</li> <li>для H-систем: односторонние I/O резервного CPU недоступны (напр., резервный CPU в режиме STOP).</li> </ul>	Только в случае SFC58 "WR_REC"
80A2	Ошибка протокола DP на уровне 2, возможно, неисправность аппаратуры.	Децентрализованная периферия
80A3	Ошибка протокола DP в случае преобразователя данных с прямым каналом передачи данных или пользовательского интерфейса/пользователя. Возможно, неисправность аппаратуры.	Децентрализованная периферия
80A4	Ошибка связи в коммуникационной шине	Ошибка происходит между CPU и внешним интерфейсным модулем DP.
80B0	<ul style="list-style-type: none"> <li>SFC недопустим для типа модуля.</li> <li>Модуль не распознает запись данных.</li> <li>Номер записи данных <math>\geq 241</math> не разрешен.</li> <li>В случае SFC58 (WR_REC) записи данных 0 и 1 не разрешены.</li> </ul>	-
80B1	В параметре RECORD задана неправильная длина.	<ul style="list-style-type: none"> <li>Для SFC58 "WR_REC": неправильная длина</li> <li>Для SFC59 "RD_REC" (только при использовании старых FM S7-300 и CP S7-300): длина &gt; длины записи</li> <li>Для SFC13 "DPNRM_DG": длина &lt; длины записи</li> </ul>
80B2	Сконфигурированный слот не занят.	-
80B3	Фактический тип модуля не совпадает с типом модуля, требуемым в SDB1.	-
80C0	В случае SFC59 (RD_REC): Модуль имеет запись данных, но все еще нет данных, которые нужно считывать. В случае SFC13 (DPNRM_DG): Нет доступных диагностических данных.	Только в случае SFC59 (RD_REC) или для SFC13 "DPNRM_DG"
80C1	Данные предыдущего задания на запись для той же самой записи данных в модуле еще не были обработаны модулем.	-
80C2	Модуль в настоящее время обрабатывает максимально возможное для CPU количество заданий.	-
80C3	Требуемые ресурсы (память и т.д.) в настоящее время заняты.	-

Код ошибки (W#16#...)	Объяснение	Ограничение
80C4	Внутренняя нерегулярная ошибка. Функция не может быть выполнена. <ul style="list-style-type: none"><li>Повторите запуск задачи. При частом повторении ошибки проверьте Вашу систему на предмет электрических помех.</li></ul>	-
80C5	Децентрализованная периферия недоступна.	Децентрализованная периферия
80C6	Передача записи данных была остановлена вследствие прерывания класса приоритета (горячий рестарт или фоновый режим).	Децентрализованная периферия
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL	



## 7.9 Чтение записи данных с помощью SFC59 "RD\_REC" в CPU S7-300

### Применимость

Следующее описание SFC59 "RD\_REC" применимо к CPU, перечисленным ниже:

CPU	Заказной номер
CPU 312 IFM	6ES7312-5AC00-0AB0
CPU 313	6ES7313-1AD00-0AB0
CPU 314	6ES7314-1AE01-0AB0
CPU 314 IFM	6ES7314-5AE00-0AB0
CPU 315	6ES7315-1AF00-0AB0
CPU 315-2DP	6ES7315-2AF00-0AB0
CPU 614	6ES7614-1AH00-0AB3

### Описание

С помощью SFC59 "RD\_REC" (read data record [читать запись данных]) Вы считываете запись данных с номером RECNUM из адресуемого модуля. Считываемая запись данных вводится в область назначения, указанную параметром RECORD, если передача данных была свободна от ошибок.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: Запрос на чтение.
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор (ID) адресной области: V#16#54 = Периферийный вход (PI) V#16#55 = Периферийный выход (PQ) Если модуль смешанный, то задайте ID области с наименьшим адресом. Если адреса одинаковые, то задайте V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес модуля. В случае смешанных модулей задайте меньший из двух адресов.
RECNUM	INPUT	BYTE	I, Q, M, D, L, константа	Номер записи данных (разрешенные значения от 0 до 240).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время выполнения функции происходит ошибка, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Чтение еще не закончилось.
RECORD	OUTPUT	ANY	I, Q, M, D, L	Область назначения для считываемой записи данных. В случае асинхронного выполнения SFC59 убедитесь, что фактические параметры RECORD имеют одинаковую информацию о длине во всех вызовах. Разрешен только тип данных BYTE.

## RECORD

Информация о длине в выходном параметре RECORD интерпретируется следующим образом:

Длина данных, которые нужно считывать из выбранной записи данных. Это означает, что информация о длине в RECORD должна быть не больше, чем фактическая длина записи данных.

Целесообразно выбирать длину для RECORD точно такую же, как фактическая длина записи данных.

### Принцип передачи данных

В случае задания на чтение CPU информирует адресованный модуль о длине параметра RECORD. Следующие пункты зависят от того, принадлежит ли модуль станции DP или нет:

- Модуль находится в центральной стойке или стойке расширения.  
Если длина, заданная посредством RECORD, короче фактической длины требуемой записи данных, то CPU считывает столько байтов от начала записи данных, сколько задано в информации о длине RECORD, и вводит их в RECORD. RET\_VAL имеет значение 0.  
Если длина в RECORD больше фактической длины требуемой записи данных, то CPU вводит в RET\_VAL код ошибки.  
Если информация о длине в RECORD такая же, как фактическая длина требуемой записи данных, то CPU считывает требуемую запись данных и вводит ее в RECORD. В RET\_VAL вводится значение 0.
- Модуль находится в slave-устройстве DP S7.  
Коммуникационный процессор ведомые DP-устройства S7 проверяет информацию о длине, полученную от CPU.  
Если длина в RECORD меньше длины требуемой записи данных, то ведомое DP-устройство S7-300 возвращает в CPU требуемую часть выбранной записи данных.  
Если длина в RECORD больше длины требуемой записи данных, то ведомое DP-устройство S7-300 возвращает в CPU информацию об ошибке. CPU проверяет информацию об ошибках или информацию о длине, принятой от ведомые DP-устройства S7:  
Если ведомое DP-устройство S7 предоставляет данные об ошибке, то соответствующий код ошибки вводится в RET\_VAL.  
Если ведомое DP-устройство S7 возвращает длину прочитанных данных, то эта длина сравнивается с информацией о длине в RECORD. В зависимости от результата сравнения в выходных параметрах RET\_VAL и RECORD делается запись. (Реакция такая же, как в случае, когда модуль расположен в центральной стойке или стойке расширения.)

---

### Примечание

В случае асинхронной обработки SFC59 убедитесь, что фактические параметры RECORD имеют одинаковую информацию о длине во всех вызовах.

---

## Информация об ошибках

Код ошибки (W#16#...)	Объяснение	Ограничение
7000	Первый вызов с REQ=0: Передача данных не активна; BUSY имеет значение 0.	-
7001	Первый вызов с REQ=1: Передача данных не активна; BUSY имеет значение 1.	Децентрализованная периферия
7002	Промежуточный вызов (REQ не имеет значения): Передача данных уже активна; BUSY имеет значение 1.	Децентрализованная периферия
8090	Задан недействительный логический базовый адрес: Нет назначения в SDB1/SDB2x или нет базового адреса.	-
8093	Этот SFC не разрешен для модуля, определяемого LADDR и IOID (разрешены следующие модули: модули S7-300 и модули S7-300 DP).	-
80A0	Отрицательное подтверждение при чтении из модуля (модуль был снят во время задания на чтение чтения или неисправен).	-
80A2	Ошибка протокола DP на уровне 2	Децентрализованная периферия
80A3	Ошибка протокола DP в случае пользовательского интерфейса/пользователя	Децентрализованная периферия
80A4	Ошибка связи в коммуникационной шине	Ошибка происходит между CPU и внешним интерфейсным модулем DP.
80B0	SFC недопустима для типа модуля. Модуль не распознает запись данных. Номер записи данных $\geq 241$ не разрешен.	-
80B1	В параметре RECORD задана неправильная длина	Длина > длины записи
80B2	Сконфигурированный слот не занят.	-
80B3	Фактический тип модуля не соответствует типу модуля, требуемому в SDB1.	-
80C0	Модуль имеет запись данных, но все еще нет данных, которые нужно считывать.	-
80C2	Модуль в настоящее время обрабатывает максимально возможное для CPU количество заданий.	-
80C3	Требуемые ресурсы (память и т.д.) в настоящее время заняты.	-
80C4	Внутренняя нерегулярная ошибка. Функция не может быть выполнена. <ul style="list-style-type: none"> <li>Повторите запуск задачи. При частом повторении ошибки проверьте Вашу систему на предмет электрических помех.</li> </ul>	-
80C5	Децентрализованная периферия недоступна.	Децентрализованная периферия
80C6	Передача записи данных была остановлена вследствие прерывания класса приоритета (горячий рестарт или фоновый режим).	Децентрализованная периферия

## 7.10 Дополнительная информация об ошибках SFC 55–59

### Только для S7-400

У S7-400 SFC 55–59 могут возвращать также информацию об ошибке W#16#80Fх. Это значит, что произошла ошибка, которую невозможно было локализовать. В этом случае обращайтесь в отдел обслуживания.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Нет ошибки
8080	Неправильный номер счетчика рабочего времени
8081	Переполнение счетчика рабочего времени
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## **8 DPV1 SFB стандарта PNO AK 1131**

## 8.1 Чтение записи данных из ведомого DP-устройства с помощью SFB52 "RDREC"

---

### Примечание

Интерфейс SFB52 идентичен FB "RDREC", определенному стандартом "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3" ("Принципы PROFIBUS в коммуникациях PROFIBUS и Proxy FB в соответствии со стандартом IEC 61131-3").

---

### Описание

С помощью SFB52 "RDREC" (read record [читать запись]) Вы можете считать запись данных с номером INDEX из ведомого компонента DP (один или несколько модулей), который может быть адресован посредством ID.

Определите в параметре MLEN максимальное число байтов для считывания информации. Если выходной параметр VALID имеет значение TRUE (ИСТИНА), то запись данных успешно считана в целевую область RECORD. В таком случае выходной параметр LEN возвращает значение длины выбранных данных в байтах.

Выходной параметр ERROR показывает, были ли ошибки при передаче записи данных. Если были ошибки, то выходной параметр STATUS содержит информацию об ошибках.

### Принцип работы

SFB52 "RDREC" выполняется асинхронно, что означает, что выполнение может перекрывать несколько вызовов. Запуск задания производится с REQ = 1.

Состояние (status) задания отображается посредством выходного параметра BUSY и байтов 2 и 3 выходного параметра STATUS. Байты 2 и 3 параметра STATUS соответственно связаны с выходным параметром RET\_VAL асинхронно работающих SFC (см. также значения REQ, RET\_VAL и BUSY асинхронно работающих SFC).

Передача записи данных завершена, если выходной параметр BUSY = FALSE (ЛОЖЬ).

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: Запрос на передачу записи данных
ID	INPUT	DWORD	I, Q, M, D, L, константа	Логический адрес ведомого DP-компонента (модуля).
INDEX	INPUT	INT	I, Q, M, D, L, константа	Номер записи данных.
MLEN	INPUT	INT	I, Q, M, D, L, константа	Максимальная длина в байтах записи данных для выборки.
VALID	OUTPUT	BOOL	I, Q, M, D, L	Новая запись данных была корректно принята.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Чтение еще не закончилось.
ERROR	OUTPUT	BOOL	I, Q, M, D, L	ERROR = 1: Произошла ошибка чтения.
STATUS	OUTPUT	DWORD	I, Q, M, D, L	ID вызова (байты 2 и 3) или код ошибки
LEN	OUTPUT	INT	I, Q, M, D, L	Длина в байтах выбранной записи данных.
RECORD	IN_OUT	ANY	I, Q, M, D, L	Область назначения для считываемой записи данных.

### Информация об ошибках

См. раздел "Получение прерывания от ведомого DP-устройства посредством SFB54 "RALRM".

## 8.2 Внесение записи данных в ведомый DP-компонент с помощью SFB53 "WRREC"

---

### Примечание

Интерфейс SFB53 идентичен FB "WRREC", определенному стандартом "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3" ("Принципы PROFIBUS в коммуникациях PROFIBUS и Proxy FB в соответствии со стандартом IEC 61131-3").

---

### Описание

С помощью SFB53 "WRREC" (write record [сделать запись]) Вы можете сделать запись данных с номером INDEX в ведомый компонент DP (модуль), который может быть адресован посредством ID.

Определите длину в байтах для передачи записи. Выбранная длина в исходной области RECORD будет составлять по крайней мере размер в байтах LEN.

Значение TRUE (ИСТИНА) выходного параметра DONE означает, что запись данных была успешно передана в ведомый компонент DP.

Выходной параметр ERROR показывает, были ли ошибки при передаче записи данных. Если были ошибки, то выходной параметр STATUS содержит информацию об ошибках.

### Принцип работы

SFB53 "WRREC" выполняется асинхронно, что означает, что выполнение может перекрывать несколько вызовов. Запуск задания производится с REQ = 1.

Состояние (status) задания отображается посредством выходного параметра BUSY и байтов 2 и 3 выходного параметра STATUS. Байты 2 и 3 параметра STATUS соответственно связаны с выходным параметром RET\_VAL асинхронно работающих SFC (см. также значения REQ, RET\_VAL и BUSY асинхронно работающих SFC).

Учтите, что Вы должны назначить одинаковые значения для фактического параметра RECORD для всех вызовов SFB53, относящихся одному и тому же заданию. Это же относится к значениям параметра LEN.

Передача записи данных завершена, если выходной параметр BUSY = FALSE (ЛОЖЬ).



Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ = 1: Запрос на передачу записи данных
ID	INPUT	DWORD	I, Q, M, D, L, константа	Логический адрес ведомого DP-компонента (модуля).
INDEX	INPUT	INT	I, Q, M, D, L, константа	Номер записи данных.
LEN	INPUT	INT	I, Q, M, D, L, константа	Максимальная длина в байтах записи данных для передачи.
DONE	OUTPUT	BOOL	I, Q, M, D, L	Запись данных была корректно передана.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Передача записи еще не закончилось.
ERROR	OUTPUT	BOOL	I, Q, M, D, L	ERROR = 1: Произошла ошибка передачи записи.
STATUS	OUTPUT	DWORD	I, Q, M, D, L	ID вызова (байты 2 и 3) или код ошибки
RECORD	IN_OUT	ANY	I, Q, M, D, L	Запись данных.

### Информация об ошибках

См. раздел "Получение прерывания от ведомого DP-устройства посредством SFB54 "RALRM".

### 8.3 Получение прерывания от ведомого DP-устройства посредством SFB54 "RALRM"

---

**Примечание**

Интерфейс SFB54 идентичен FB "RALRM", определенному стандартом "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3" ("Принципы PROFIBUS в коммуникациях PROFIBUS и Proxy FB в соответствии со стандартом IEC 61131-3").

---

**Описание**

SFB "RALRM" получает прерывание со всей соответствующей информацией от периферийного модуля (централизованная структура) или от ведомого компонента DP. Эта информация помещается в его выходные параметры.

Информация в выходных параметрах содержит стартовую информацию вызываемого OB, также как и информацию источника прерывания.

Вызывайте SFB54 только в OB прерывания, запускаемого операционной системой CPU в результате обработки прерывания ввода-вывода.

---

**Примечание**

Если Вы вызываете SFB 54 "RALRM" в OB, стартовое событие для которого было инициировано не периферией, SFB помещает соответственно "урезанную" информацию в выходные параметры.

Обеспечьте использование различных экземпляров DB при вызове SFB 54 "RALRM" в разных OB. Более того, если Вы хотите проверять данные, полученные при вызове SFB 54 для несвязанного OB прерывания, Вы должны использовать отдельный экземпляр DB для OB стартового события.

---

**Вызов SFB 54**

Вы можете вызывать SFB 54 "RALRM" в трех режимах работы (MODE), показанных в следующей таблице.

Режим	SFB 54
0	... показывает на компонент, который инициировал прерывание, в выходном параметре ID и устанавливает выходной параметр NEW в состояние TRUE (ИСТИНА).
1	... описывает все выходные параметры, независимо от компонента, который инициировал прерывание.
2	... проверяет, действительно ли компонент, определенный во входном параметре F_ID, инициировал прерывание: <ul style="list-style-type: none"> <li>• если нет, то параметр NEW = FALSE (ЛОЖЬ);</li> <li>• если да, то параметр NEW = TRUE (ИСТИНА), и другие выходные параметры будут также описаны.</li> </ul>

Параметр	Описание	Тип данных	Область памяти	Характеристика
MODE	INPUT	INT	I, Q, M, D, L, константа	Режим работы.
F_ID	INPUT	DWORD	I, Q, M, D, L, константа	Логический начальный адрес DP-компонента (модуля), инициировавшего прерывание.
MLEN	INPUT	INT	I, Q, M, D, L, константа	Максимальная длина в байтах получаемых данных прерывания.
NEW	OUTPUT	BOOL	I, Q, M, D, L	Новое прерывание было получено.
STATUS	OUTPUT	DWORD	I, Q, M, D, L	Код ошибки от SFB или ведущего DB-устройства.
ID	OUTPUT	DWORD	I, Q, M, D, L	Логический начальный адрес компонента (модуля), инициировавшего прерывание. Старший бит содержит ID для I/O: 0 – для входного адреса, 1 – для выходного адреса.
LEN	OUTPUT	INT	I, Q, M, D, L	Длина в байтах полученных данных прерывания.
TINFO	IN_OUT	ANY	I, Q, M, D, L	(информация задания) Область назначения для запуска OB информации управления.
AINFO	IN_OUT	ANY	I, Q, M, D, L	(информация прерывания) Область назначения для информации заголовка и дополнительной информации прерывания. Для AINFO Вы должны обеспечить область с размерами по крайней мере MLEN байтов.

**Внимание**

Если Вы выбрали слишком малую область назначения TINFO или AINFO, то не вся информация, полученная SFB 54, будет записана.

### Структура данных для области назначения TINFO

Байты	Значение
0 ... 19	Стартовая информация ОВ, из которого производится текущий вызов SFC 54.
20 ... 27	Информация управления.

### Структура информации управления

Байт № для TINFO	Тип данных	Значение			
20	BYTE	центральная:	0		
		распределенная:	ID ведущего DP-устройства (возможные значения: 0...255)		
21	BYTE	центральная:	Номер модуля стойки (возможные значения: 0...31)		
		распределенная:	Номер DP-станции (возможные значения: 0...127)		
22	BYTE	центральная:	0		
		распределенная:	• биты 0...3	тип Slave (ведомый)	0000: DP 0001: DPS7 0010: DPS7 V1 0011: DPV1 0100: резерв
			• биты 4...7	тип Profile	0000: DP 0001: резерв
23	BYTE	центральная:	0		
		распределенная:	• биты 0...3	тип Interrupt info (инф-я прерывания)	0000: Transparent (Явная) (Прерывание от сконфигурированного распределенного модуля)
					0001: Представительная (Прерывание не из DPV1 ведомого устройства или из не-сконфигурированного слота)

## Структура информации управления

(продолжение)

Байт № для TINFO	Тип данных	Значение				
					0010:	Сгенерированная (прерывание, генерировано в CPU)
					0011:	Резерв
			• биты 4...7	Structure version (версия структуры)	0000:	Исходная
					0001:	Резерв
24	BYTE	центральная:	0			
		распределенная	Флаги интерфейса ведущего DP-устройства			
			• бит 0=0	Прерывание от интегрированного DP-интерфейса		
			• бит 0=1	Прерывание от внешнего DP-интерфейса		
			• биты 1...7	резерв		
25	BYTE	центральная:	0			
		распределенная	Флаги интерфейса ведомого DP-устройства			
			• бит 0	Содержимое бита EXT_DIAG_Bit структуры сообщения диагностики или 0, если этот бит не существует.		
			• биты 1...7	резерв		
26, 27	WORD	центральная:	0			
		распределенная	ID-номер PROFIBUS			

## Структура данных области назначения AINFO

Байт	Значение	
0...3	Информация заголовка:	
4...223	Дополнительная информация прерывания: специальные данные модуля для соответствующего прерывания:	
	центральная:	Элементы массива ARRAY[0] ... ARRAY[220]
	распределенная:	Элементы массива ARRAY[0] ... ARRAY[59]

## Структура данных заголовка Header Information

Байт	Тип данных	Значение		
0	BYTE	Длина информации для соответствующего прерывания в байтах		
		центральная:	1...224	
		распределенная:	4...63	
1	BYTE	центральная:	Резерв	
		распределенная:	ID соответствующего типа прерывания	
			1:	Прерывание диагностики
			2:	Аппаратное прерывание
			3:	Прерывание удаления
			4:	Прерывание вставки
			5:	Прерывание статуса
			6:	Прерывание обновления (модификации)
			32...126	Прерывание определенное изготовителем
2	BYTE	Номер слота компонента, вызвавшего прерывание		
3	BYTE	центральная:	Резерв	
		распределенная:	Спецификация	
			биты 0 и 1	0: больше нет информации 1: входящее событие, прерывание на слоте 2: исходящее событие, прерывание на слоте завершено 3: исходящее событие, прерывание на слоте все еще активно
			бит 2:	Add_Ack (подтверждение)
			биты 3...7:	Порядковый номер

## Области назначения TINFO и AINFO

Степень заполнения областей назначения TINFO и AINFO зависит от ОВ, в котором вызывается функция SFB54. См. представленную ниже таблицу с данными по заполнению областей назначения TINFO и AINFO.

Тип прерывания	ОВ	TINFO информация состояния ОВ	TINFO информация управления	AINFO информация заголовка	AINFO дополнительная информация прерывания	
					центральная:	распределенная:
Аппаратное прерывание	4x	Да	Да	Да	Нет	Нет
						как выдает ведомое DP-устройство
Прерывание состояния	55	Да	Да	Да	Да	
Прерывание модификации	56	Да	Да	Да	Да	
Прерывание производителя	57	Да	Да	Да	Да	
Прерывание при ошибке избыточности периферии	70	Да	Да	Нет	Нет	
Прерывание диагностики	82	Да	Да	Да	Нет	Запись данных 1
						как выдает ведомое DP-устройство
Прерывание удаления/вставки	83	Да	Да	Да	Нет	Нет
						как выдает ведомое DP-устройство
Прерывание при сбое модуля стойки/станции	86	Да	Да	Нет	Нет	
...	все прочие ОВ	Да	Нет	Нет	Нет	

### Информация об ошибках

Выходной параметр STATUS содержит информацию об ошибках. Он состоит из 4-байтового массива ARRAY[1...4] OF BYTE и имеет структуру, показанную в нижеследующей таблице:

Элемент массива	Имя	Значение
STATUS[1]	Function_Num	<ul style="list-style-type: none"> <li>В#16#00, если нет ошибки</li> <li>ID функции из DPV1-PDU: в случае ошибки В#16#80 подключается с функцией OR. Если не используется протокол DPV1, В#16#C0.</li> </ul>
STATUS[2]	Error Decode	Адрес ID ошибки
STATUS[3]	Error_Code_1	ID ошибки
STATUS[4]	Error_Code_2	Расширенная информация по ID ошибки производителя

STATUS[2] может иметь следующие значения:

Error Decode (В#16#...)	Источник	Значение
00...7F	CPU	Нет ошибок или нет предупреждений
80	DPV1	Ошибка согласно IEC 61158-6
81...8F	CPU	В#16#8x указывает на ошибку в n-ном параметре вызова SFB
FE, FF	DP Profile	Profile-ошибка

STATUS[3] может иметь следующие значения:

Error Decode (В#16#...)	Error_Code_1 (В#16#...)	Объяснение в соответствии с DVP1	Значение
00	00		нет ошибок, нет предупреждений
70	00	резервируется, отбрасывается	начальный вызов: нет передачи записи данных
	01	резервируется, отбрасывается	начальный вызов: передача записи данных началась
	02	резервируется, отбрасывается	промежуточный вызов: ведется передача записи данных



(продолжение таблицы)

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Объяснение в соответствии с DVP1	Значение
80	90	резервируется, проходит	некорректный начальный адрес
	92	резервируется, проходит	некорректный тип указателя ANY
	93	резервируется, проходит	DP-компонент, адресуемый посредством ID или F_ID неконфигурирован
	96	CPU	Произошло переключение "ведущий – запасной" в H системе, и ранее активный ведущий CPU перешел в режим STOP. В то же время обрабатывался OB. SFB 54 не может выдать стартовую информацию OB, информацию управления, информацию заголовка или информацию дополнительного прерывания. Вы можете считать стартовую информацию OB с помощью SFC 6 "RD_SINFO". Кроме того, Вы можете использовать SFC 13 "DPNRM_DG" для синхронного чтения текущей диагностической информации от поврежденного ведомого DP-устройства для блоков OB 4x, 55, 56, 57, 82 и 83 (Адресация из стартовой информации OB).
	A0	ошибка чтения	Не подтверждается чтение в модуле
	A1	ошибка записи	Не подтверждается запись в модуле
	A2	сбой модуля	Ошибка протокола DP в уровне, возможен сбой оборудования
	A3	резервируется, проходит	Ошибка протокола DP в DDLM или в уровне интерфейс пользователя-пользователь, возможен сбой оборудования
	A4	резервируется, проходит	Сбой в шине связи
	A5	резервируется, проходит	-
	A7	резервируется, проходит	Ведомое устройство DP или модули заняты (временная ошибка)
	A8	несовпадение версии	Ведомое устройство DP или модули сообщают о несовпадении версий
	A9	характеристика не поддерживается	Ведомое устройство DP или модули не поддерживают характеристику
	AA ... AF	определяется пользователем	Ведомое устройство DP или модули сообщают об определенной производителем ошибке в приложении. Обратитесь к документации по ведомому DP-устройству или модулям.

(продолжение таблицы)

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Объяснение в соответствии с DVP1	Значение
	B0	неверный индекс	Запись данных, не известная в модуле Запрещенный номер =>256 записи данных
	B1	ошибка размера записи	Неправильная длина указана в параметре RECORD ; для SFB54: ошибка длины в AINFO
	B2	ошибочный слот	Конфигурированный слот свободен
	B3	конфликт типов	Фактический тип модуля не соответствует определению
	B4	ошибка области	Ведомое устройство DP или модуль сообщают о доступе к неправильно заданной области
	B5	ошибка состояния	Ведомое устройство DP или модуль не в состоянии "Готов"
	B6	ошибка доступа	Ведомое устройство DP или модуль не обеспечивают доступ
	B7	ошибка диапазона	Ведомое устройство DP или модуль сообщают о выходе параметра (сигнала) за разрешенный диапазон
	B8	ошибка параметра	Ведомое устройство DP или модуль сообщают об ошибочном параметре
	B9	ошибка типа	Ведомое устройство DP или модуль сообщают об ошибочном типе
	BA ... BF	определяется пользователем	Ведомое устройство DP или модуль сообщают об определенной производителем ошибке доступа. Обратитесь к документации по ведомому DP-устройству или модулю.
	C0	конфликт, возникший при чтении	В блоке имеется запись данных, однако, нет никаких считанных данных.
	C1	конфликт, возникший при записи	Данные предыдущего запроса записи к модулю для той же самой записи данных еще не были обработаны модулем.
	C2	ресурс занят	В наст. время модуль обрабатывает максимально возможное число заданий для CPU.
	C3	ресурс недоступен	Требуемый для выполнения задания ресурс занят.
	C4		Внутренняя временная ошибка. Задание не может быть выполнено. Повторите задание. При частом появлении подобной ошибки проверьте Вашу установку на наличие источника электрических помех.
	C5		Ведомое устройство DP или модуль не доступен.

(продолжение таблицы)

Error Decode (B#16#...)	Error_Code_1 (B#16#...)	Объяснение в соответствии с DVP1	Значение
	C6		Передача записи данных отменена из-за отмены (потери) приоритетного класса
	C7		Выполнение задания отменено из-за рестарта (теплого рестарта) или холодного рестарта ведущего DP-устройства.
	C8 ... CF		Ведомое устройство DP или модуль сообщают об определенной производителем ошибке доступа к ресурсу. Обратитесь к документации по ведомому DP-устройству или модулю.
	Dx	определяется пользователем	Параметр ведомого устройства DP. Обратитесь к документации по ведомому DP-устройству.
81	00 ... FF		Ошибка при первом вызове параметра (для SFB54: MODE).
	00		Неразрешенный рабочий режим.
82	00 ... FF		Ошибка при втором вызове параметра
:	:		:
88			Ошибка при восьмом вызове параметра (для SFB54: TINFO)
	01		Синтаксическая ошибка в ID.
	23		Превышено количество в группе данных или область назначения слишком мала.
	24		Ошибка диапазона ID.
	32		Номер DB/DI за пределами диапазона пользователя.
	3A		Номер DB/DI равен NULL для области ID DB/DI или определяемый DB/DI не существует.
89	00 ... FF		Ошибка при девятом вызове параметра (для SFB54: AINFO)
	01		Синтаксическая ошибка в ID.
	23		Превышено количество в группе данных или область назначения слишком мала.
	24		Ошибка диапазона ID.
	32		Номер DB/DI за пределами диапазона пользователя.
	3A		Номер DB/DI равен NULL для области ID DB/DI или определяемый DB/DI не существует.

*(продолжение таблицы)*

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Объяснение в соответствии с DVP1	Значение
8A	00 ... FF		Ошибка в 10 вызове параметра
:	:		:
8F	00 ... FF		Ошибка в 15 вызове параметра
FE, FF	00 ... FF		Profile-ошибка

При DPV1-ошибке ведущее DP-устройство передает CPU и SFB параметр состояния STATUS[4]. При отсутствии DPV1-ошибки значение параметра равно 0 при следующих исключениях для SFB52:

- STATUS[4] содержит размер области назначения от RECORD, если MLEN больше размера области назначения от RECORD.
- STATUS[4] = MLEN, если фактическая длина записи данных меньше MLEN, а MLEN меньше размера области назначения от RECORD.

## **9 SFC для управления прерываниями по времени**

## 9.1 Манипулирование прерываниями по времени

### Определение

Прерывание по времени имеет сопровождается вызовом ОВ прерываний по времени (ОВ10 – ОВ17).

### Условия для вызова

Прежде чем операционной системой может быть вызван ОВ прерываний по времени, должны быть выполнены следующие условия:

- ОВ прерываний по времени должен быть параметрирован (дата и время запуска, выполнение) с помощью
  - STEP 7 или
  - SFC28 "SET\_TINT" в программе пользователя.
- ОВ прерываний по времени должен быть активирован с помощью
  - STEP 7 или
  - SFC30 "ACT\_TINT" в программе пользователя.
- Выбор ОВ прерываний по времени не должен быть отменен с помощью STEP 7.
- ОВ прерываний по времени должен существовать в CPU.
- Если Вы устанавливаете прерывание по времени с помощью SFC30 "ACT\_TINT" и если Вы задали исполнение этого ОВ как **однократное**, то дата и время запуска еще не должны пройти. Если Вы выбрали **периодическое** выполнение, то ОВ прерываний по времени будет вызван, когда будет завершен следующий период (момент времени запуска + кратное длительности периода).

### Совет

Вы можете назначать параметры прерыванию по времени с помощью STEP 7, а затем активировать прерывание в свое пользовательской программе (SFC30 "ACT\_TINT").

### Назначение SFC28 – SFC31

Системные функции SFC28 – SFC31, описанные в следующих разделах, используются следующим образом:

- для установки прерываний по времени (SFC28 "SET\_TINT")
- для отмены прерываний по времени (SFC29 "CAN\_TINT")
- для активации прерываний по времени (SFC30 "ACT\_TINT")
- для опроса прерываний по времени (SFC31 "QRY\_TINT")

## 9.2 Характеристики SFC 28 ... 31

### Что произойдет, если...

Следующая таблица перечисляет ряд различных ситуаций и объясняет, какое воздействие они оказывают на прерывание по времени.

Если ...	то ...
прерывание по времени устанавливается (вызовом SFC28; SET_TINT)	текущее прерывание по времени автоматически отменяется.
прерывание по времени отменяется (вызовом SFC29; CAN_TINT)	дата и время запуска стираются. Затем прерывание по времени должно быть сначала вновь установлено, прежде чем оно может быть активировано.
ОВ прерываний по времени не существует в момент вызова	автоматически генерируется ошибка класса приоритета; т.е. операционная система вызывает ОВ 85. Если ОВ 85 не существует, то CPU переходит в состояние STOP.
синхронизируются часы реального времени или корректируется время <ul style="list-style-type: none"> <li>Часы переводятся вперед</li> </ul>	Если дата/время запуска пропущены из-за перевода часов вперед: <ul style="list-style-type: none"> <li>Операционная система вызывает ОВ80<sup>1</sup>.</li> <li>Вслед за ОВ80 вызываются все пропущенные ОВ прерываний по времени (один раз, независимо от количества пропущенных периодов), если они не подвергались обработке в ОВ80<sup>2</sup>.</li> </ul> Если ОВ 80 не существует, то CPU переходит в состояние STOP.
<ul style="list-style-type: none"> <li>Часы переводятся назад</li> </ul>	Если ОВ прерывания по времени уже вызывались на интервале переведенного назад времени, то при новом прохождении этого времени они еще раз не вызываются.

- 1) В информации о событии запуска ОВ 80 кодируется то, какие ОВ прерывания по времени не могут вызываться из-за установки времени вперед. Время в информации о событии запуска соответствует установленному вперед времени.
- 2) Время в информации о событии запуска прерывания по времени, активируемого позднее, после того как оно было пропущено, соответствует времени запуска первого пропущенного прерывания по времени.

### Теплый рестарт и холодный рестарт

Во время теплого или холодного рестарта все настройки прерываний по времени, сделанные в программе пользователя с помощью SFC, стираются.

После этого действуют установленные посредством STEP 7 параметры из блока параметров "time-of-day interrupts [прерывания по времени]".

**Выполнение ОВ прерываний по времени**

Следующая таблица показывает различные воздействия параметра "execution [выполнение]". Этот параметр устанавливается с помощью STEP 7 или с помощью SFC 28 "SET\_TINT" (входной параметр PERIOD).

Выполнение ОВ прерываний по времени	Реакция
Нет (можно устанавливать только с помощью STEP 7)	ОВ прерываний по времени не выполняется даже тогда, когда он существует в CPU. Параметры могут быть заново установлены в программе пользователя с помощью SFC 28 "SET_TINT" (установить прерывание по времени).
Однократно	Прерывание по времени после вызова ОВ прерываний по времени отменяется. Затем оно может быть вновь установлено и активировано.
Периодически (ежеминутно, ежечасно, ежедневно, еженедельно, ежемесячно, ежегодно)	Если дата и время запуска к моменту активации уже прошли, то ОВ прерываний по времени прерывает циклическую программу в момент времени "дата/время запуска + кратное установленной длительности периода". В очень редких случаях ОВ прерываний по времени может при следующем вызове оказаться еще в состоянии обработки. Результат: <ul style="list-style-type: none"> <li>• ошибка времени (операционная система вызывает ОВ 80; если ОВ 80 не существует, то CPU переходит в состояние STOP).</li> <li>• ОВ прерываний по времени выполняется позднее.</li> </ul>



### 9.3 Установка прерывания по времени с помощью SFC28 "SET\_TINT"

#### Описание

С помощью SFC 28 "SET\_TINT" (set time-of-day interrupt [установить прерывание по времени]) устанавливаются дата и время запуска организационных блоков прерываний по времени. Секунды и миллисекунды в заданном времени запуска игнорируются и устанавливаются в 0.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер OB, который запускается в момент времени SDT + кратное PERIOD (OB10 – OB17).
SDT	INPUT	DT	D, L, константа	Дата и время запуска: секунды и миллисекунды в заданном времени запуска игнорируются и устанавливаются в 0.
PERIOD	INPUT	WORD	I, Q, M, D, L, константа	Периоды от исходной точки SDT: W#16#0000 = однократно W#16#0201 = ежеминутно W#16#0401 = ежечасно W#16#1001 = ежедневно W#16#1202 = еженедельно W#16#1401 = ежемесячно W#16#1801 = ежегодно W#16#2001 = в конце месяца
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.

#### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было
8090	Ошибочный параметр OB_NR
8091	Ошибочный параметр SDT
8092	Ошибочный параметр PERIOD
80A1	Установленный момент запуска лежит в прошлом.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 9.4 Отмена прерывания по времени с помощью SFC29 "CAN\_TINT"

### Описание

С помощью SFC 29 "CAN\_TINT" (cancel time-of-day interrupt [отменить прерывание по времени]) можно отменить активированный организационный блок прерываний по времени.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер OB, в котором будут отменены дата и время запуска (OB10 – OB17).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было.
8090	Ошибочный параметр OB_NR
80A0	Не установлены дата и время запуска для соответствующего OB прерываний по времени
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 9.5 Активация прерывания по времени с помощью SFC30 "ACT\_TINT"

### Описание

С помощью SFC 30 "ACT\_TINT" (activate time-of-day interrupt [активировать прерывание по времени]) Вы можете активировать организационный блок прерываний по времени.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер OB, который должен активироваться (OB10 – OB17).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было.
8090	Ошибочный параметр OB_NR.
80A0	Не установлены дата и время запуска для OB прерываний по времени.
80A1	Активированное время лежит в прошлом; ошибка имеет место только в случае, когда выбрано однократное выполнение (execution=once).
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 9.6 Опрос прерывания по времени с помощью SFC31 "QRY\_TINT"

### Описание

С помощью системной функции SFC 31 "QRY\_TINT" (query time-of-day interrupt [опросить прерывание по времени]) можно отобразить состояние организационного блока прерываний по времени в выходном параметре STATUS.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер OB, состояние которого опрашивается (OB10 – OB17).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.
STATUS	OUTPUT	WORD	I, Q, M, D, L	Состояние прерывания по времени; см. следующую таблицу.

### Выходной параметр STATUS

Бит	Значение	Объяснение
0	0	Прерывание по времени разрешено операционной системой.
1	0	Новые прерывания по времени принимаются.
2	0	Прерывание по времени не активировано или выполнено.
3	-	-
4	0	OB прерываний по времени не загружен.
5	0	Выполнение OB прерываний по времени заблокировано работающей функцией тестирования.
6	0	Отсчет для временного прерывания от базового системного времени
	1	Отсчет для временного прерывания от локального времени

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было.
8090	Ошибочный параметр OB_NR
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## **10 SFC для управления прерываниями с задержкой**

## 10.1 Манипулирование прерываниями с задержкой

### Определение

После того как Вы вызвали SFC 32 "SRT\_DINT", операционная система по истечении заданного времени задержки генерирует прерывание, иными словами, вызывается выбранный OB прерываний с задержкой. Это прерывание известно как прерывание с задержкой.

### Условия для вызова

Чтобы прерывание с задержкой могло быть вызвано операционной системой, должны быть выполнены следующие условия:

- OB прерываний с задержкой должен быть запущен посредством SFC32 "SRT\_DINT."
- Выбор OB прерываний с задержкой не должен быть отменен с помощью STEP 7.
- OB прерываний с задержкой должен существовать в CPU.

### Назначение SFC32 – SFC34

Системные функции SFC32 – SFC34, описанные в следующих разделах, используются следующим образом:

- для запуска прерываний с задержкой (SFC32 "SRT\_DINT")
- для отмены прерываний с задержкой (SFC33 "CAN\_DINT")
- для опроса прерываний с задержкой (SFC34 "QRY\_DINT").

**Что произойдет, если...**

Следующая таблица перечисляет ряд ситуаций и описывает их влияние на прерывание с задержкой.

<b>Если ...</b>	<b>и ...</b>	<b>то ...</b>
запускается прерывание с задержкой (вызовом SFC32 "SRT_DINT").	уже запущено прерывание с задержкой,	время задержки заменяется; прерывание с задержкой запускается вновь.
	ОВ прерываний с задержкой не существует к моменту вызова,	операционная система генерирует ошибку класса приоритета (вызывает OB85). Если OB 85 не существует, то CPU переходит в состояние STOP.
	запуск прерывания произошло в ОВ запуска и время задержки истекло прежде, чем CPU оказался в RUN,	вызов ОВ прерывания с задержкой задерживается до тех пор, пока CPU не перейдет в RUN.
время задержки истекло	ранее запущенный ОВ прерываний с задержкой еще исполняется,	операционная система генерирует ошибку времени (вызывает OB80 Если OB 80 не существует, то CPU переходит в состояние STOP.

**Теплый рестарт и холодный рестарт**

При теплом и холодном рестарте все настройки прерывания с задержкой, сделанные в программе пользователя с помощью SFC, стираются.

**Старт в ОВ запуска**

Прерывание с задержкой может стартовать в ОВ запуска. Для вызова ОВ прерываний с задержкой должны быть выполнены два условия:

- Истекло время задержки.
- CPU находится в режиме RUN.

Если время задержки истекло, а CPU еще не находится в режиме RUN, то вызов ОВ прерываний с задержкой откладывается до тех пор, пока CPU не перейдет в состоянии RUN. Тогда ОВ прерываний с задержкой вызывается еще до первой команды в ОВ1.

## 10.2 Запуск прерывания с задержкой с помощью SFC32 "SRT\_DINT"

### Описание

С помощью SFC 32 "SRT\_DINT" (start time–delay interrupt [запустить прерывание с задержкой]) Вы запускаете прерывание с задержкой, которое по истечении времени задержки (параметр DTIME) вызывает OB прерываний с задержкой.

С помощью параметра SIGN Вы можете ввести идентификатор для распознавания запуска прерывания с задержкой. Значения DTIME и SIGN вновь появляются в информации о событии запуска указанного OB, когда он выполняется.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер OB для запуска с задержкой времени (OB20 – OB23).
DTIME	INPUT	TIME	I, Q, M, D, L, константа	Длительность задержки (от 1 до 60000 мс)
SIGN	INPUT	WORD	I, Q, M, D, L, константа	Идентификатор, который вводится в информацию о событии запуска OB при вызове OB прерываний с задержкой.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.

### Точность

Время между вызовом SFC 32 "SRT\_DINT" и запуском OB прерываний с задержкой максимум на **одну миллисекунду** меньше, чем выбранное время, если только никакие события прерывания не задерживают вызов.

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было.
8090	Ошибочный параметр OB_NR
8091	Ошибочный параметр DTIME
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL



### 10.3 Опрос состояния прерывания с задержкой с помощью SFC34 "QRY\_DINT"

#### Описание

С помощью SFC 34 "QRY\_DINT" (query time–delay interrupt [опросить прерывание с задержкой]) Вы можете опрашивать состояние прерывания с задержкой. Прерывания с задержкой управляются организационными блоками OB20 – OB23.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер OB, состояние которого будет опрошено (OB20 – OB23).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.
STATUS	OUTPUT	WORD	I, Q, M, D, L	Состояние прерывания с задержкой, см. следующую таблицу

#### Выходной параметр STATUS

Бит	Значение	Смысл
0	0	Прерывание с задержкой разблокировано операционной системой.
1	0	Новые прерывания с задержкой не отвергнуты.
2	0	Прерывание с задержкой не активировано или выполнено.
3	-	-
4	0	OB прерываний с задержкой не загружен.
5	0	Выполнение OB прерываний с задержкой заблокировано работающей функцией тестирования.

#### Информация об ошибках

Код ошибки (W#16#..)	Описание
0000	Ошибки не было
8090	Ошибочный параметр OB_NR
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 10.4 Отмена прерывания с задержкой с помощью SFC33 "CAN\_DINT"

### Описание

С помощью SFC 33 "CAN\_DINT" (cancel time–delay interrupt [отменить прерывание с задержкой]) Вы можете отменить уже запущенное прерывание с задержкой (см. раздел "Запуск прерывания с задержкой с помощью SFC32 "SRT\_DINT"). Тогда ОВ прерываний с задержкой не вызывается.

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер ОВ, который должен быть отменен (OB20 – OB23).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении функции происходит ошибка, то фактический параметр RET_VAL содержит код ошибки.

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было.
8090	Ошибочный параметр OB_NR
8хху	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## **11 SFC для обработки синхронных ошибок**

## 11.1 Маскирование синхронных ошибок

### Введение

Синхронные ошибки – это ошибки программирования и доступа. Такие ошибки происходят в результате программирования с неправильными адресными областями, номерами или неправильными адресами.

**Маскирование** этих синхронных ошибок означает следующее:

- Маскированные синхронные ошибки не запускают ОВ ошибок и не приводят к запрограммированной альтернативной реакции.
- CPU "регистрирует" произошедшие маскированные ошибки в регистре ошибок.

Маскирование синхронных ошибок выполняется посредством вызова SFC36 "MSK\_FLT".

**Демаскирование** ошибок означает отмену предварительно установленной маски и очистку соответствующего бита в регистре состояний событий текущего класса приоритета. Маскирование отменяется следующим образом:

- посредством вызова SFC37 "DMSK\_FLT"
- когда завершится текущий класс приоритета.

Если ошибка происходит после того, как она была демаскирована, то операционная система запускает соответствующий ОВ ошибок. Вы можете запрограммировать ОВ121 для реагирования на ошибки программирования и ОВ122 для реагирования на ошибки доступа.

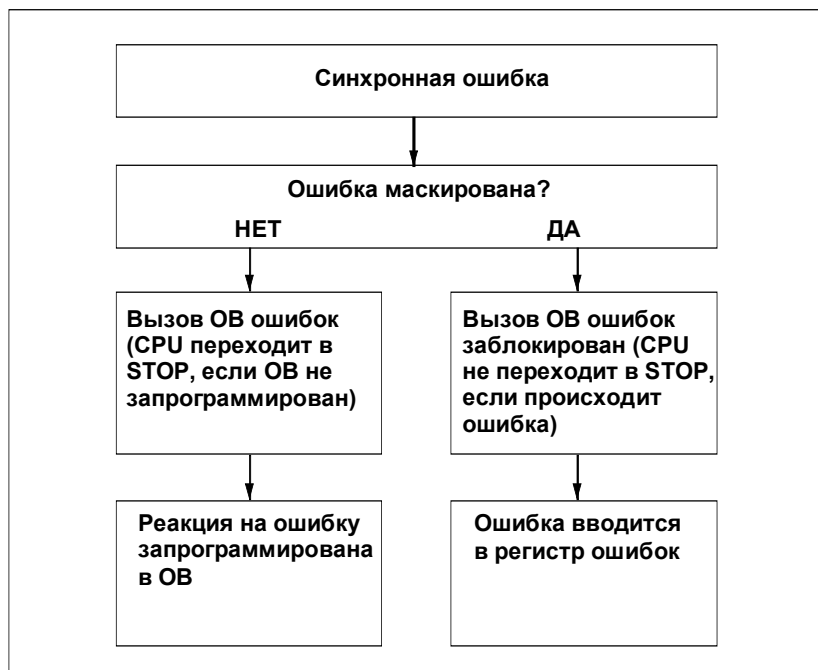
Вы можете использовать SFC38 "READ\_ERR", чтобы считывать произошедшие замаскированные ошибки.

Обратите внимание: В S7-300 (кроме CPU 318), независимо от того, замаскирована ошибка или демаскирована, она вводится в диагностический буфер, и светится светодиод групповой ошибки CPU.

### Обработка ошибок в общем случае

Если в программе пользователя происходят ошибки программирования и доступа, то Вы можете реагировать на них разными способами:

- Вы можете запрограммировать ОВ ошибок, который вызывается операционной системой, когда происходит соответствующая ошибка.
- Вы можете заблокировать вызов ОВ ошибок индивидуально для каждого класса приоритета. В этом случае, когда ошибка данного типа происходит в конкретном классе приоритета, CPU не переходит в STOP. CPU вводит ошибку в регистр ошибок. Однако из этой записи Вы не можете узнать, когда или как часто происходила ошибка.



## Фильтры

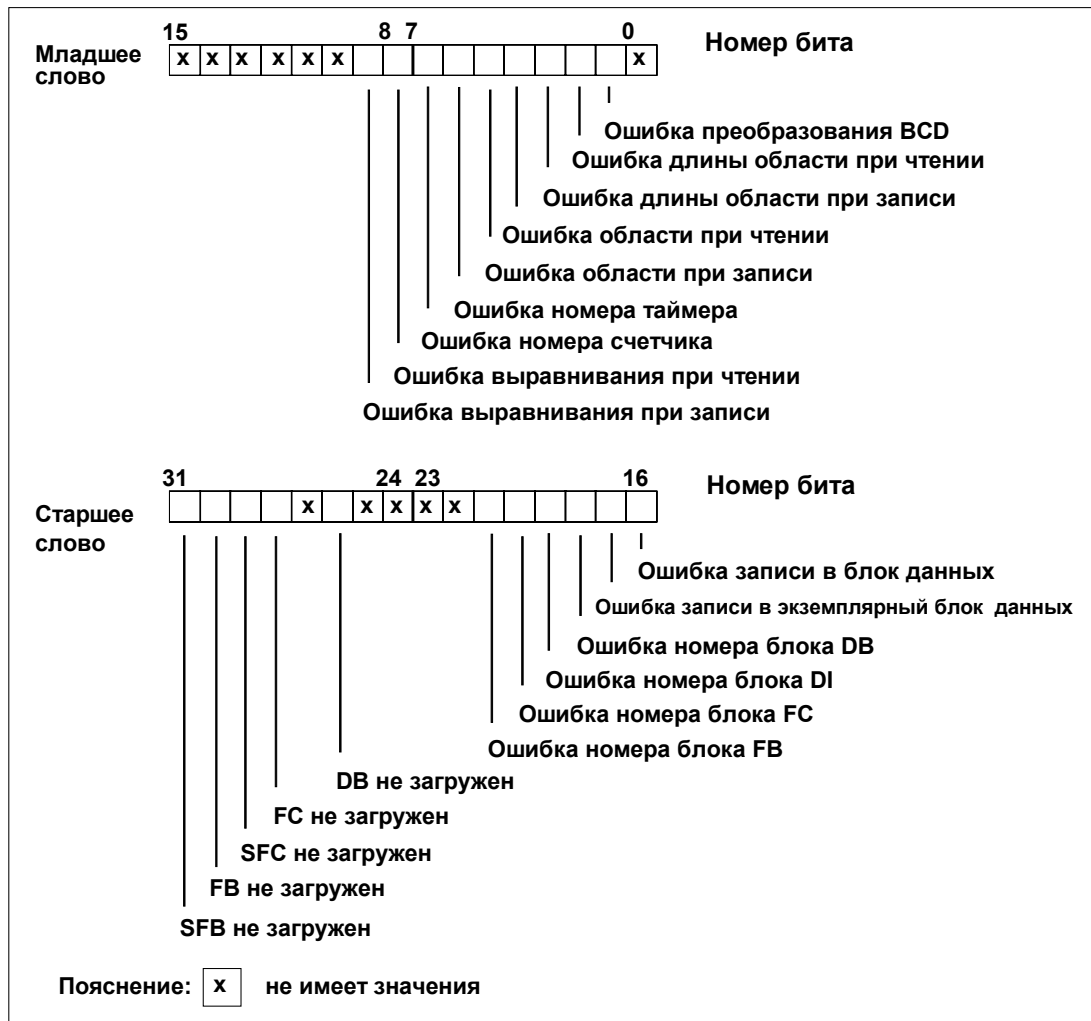
Синхронные ошибки ставятся в соответствие конкретной битовой комбинации, известной как **фильтр ошибок (маска)**. Этот фильтр ошибок находится также во входных и выходных параметрах SFC 36, 37 и 38.

Синхронные ошибки делятся на ошибки программирования и ошибки доступа, которые Вы можете маскировать, используя два фильтра ошибок. Фильтры ошибок иллюстрируются на следующих рисунках.

### Фильтр ошибок программирования

Следующий рисунок показывает битовую комбинацию фильтра ошибок программирования. Фильтр ошибок программирования расположен в параметрах PRGFLT\_...

См. также раздел "Возможные причины возникновения ошибок программирования".



#### Примечание

Биты 29 ("SFC не загружен") и 31 ("SFB не загружен") в старшем слове фильтра для ошибок программирования применимы только для S7-400 и CPU 318.

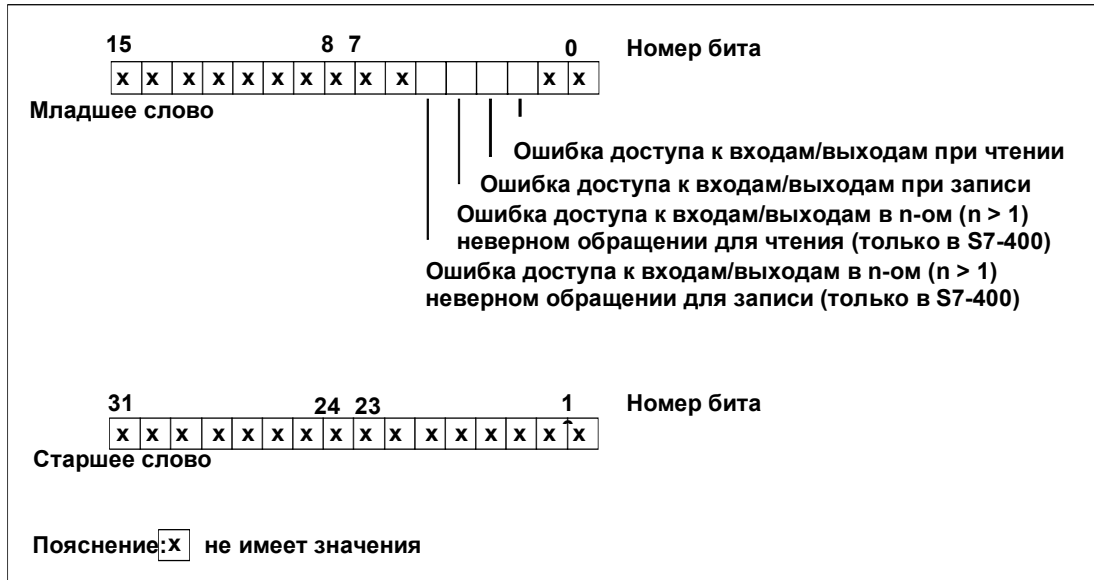
### Несущественные биты

На вышеприведенном рисунке x означает...

- ... входные параметры для SFC36, 37, 38 = "0"
- ... выходные параметры для SFC36, 37 = "1" для S7-300  
= "0" для S7-400
- для SFC38 = "0"

### Фильтр ошибок доступа для всех CPU

Следующий рисунок показывает битовую комбинацию фильтра ошибок доступа для всех CPU, кроме CPU417 и CPU 417H. Фильтр ошибок доступа находится в параметрах ACCFLT\_ ... За объяснением ошибок доступа обратитесь к таблице "Возможные причины ошибок для всех CPU, кроме CPU 31x и CPU 318" или "Возможные причины ошибок для всех CPU, кроме CPU 417 и CPU 417H".

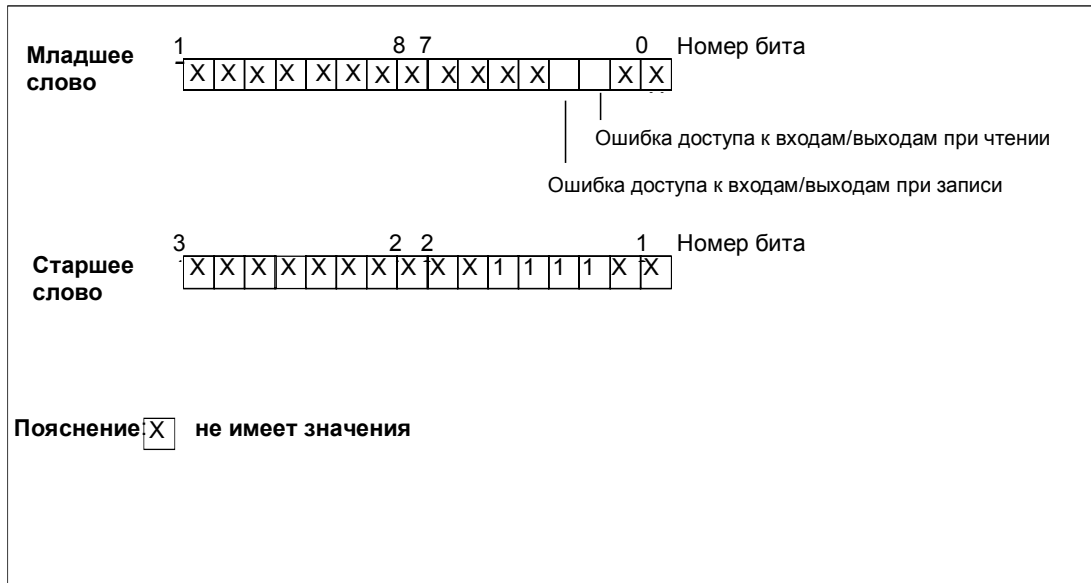


### Несущественные биты

- На вышеприведенном рисунке x означает... для SFC36, 37, 38 = "0"
- ... входные параметры для SFC36, 37 = "1" для S7-300  
= "0" для S7-400 и CPU318
- для SFC38 = "0"

### Фильтр ошибок доступа для CPU 417 и CPU 417H

Следующий рисунок показывает битовую комбинацию фильтра ошибок доступа в CPU417 и CPU 417H. Фильтр ошибок доступа находится в параметрах ACCFLT\_... За объяснением ошибок доступа обратитесь к таблице "Возможные причины ошибок для всех CPU, кроме CPU 417 и CPU 417H".



### Несущественные биты для CPU 41x и CPU 318

На вышеприведенном рисунке x означает...

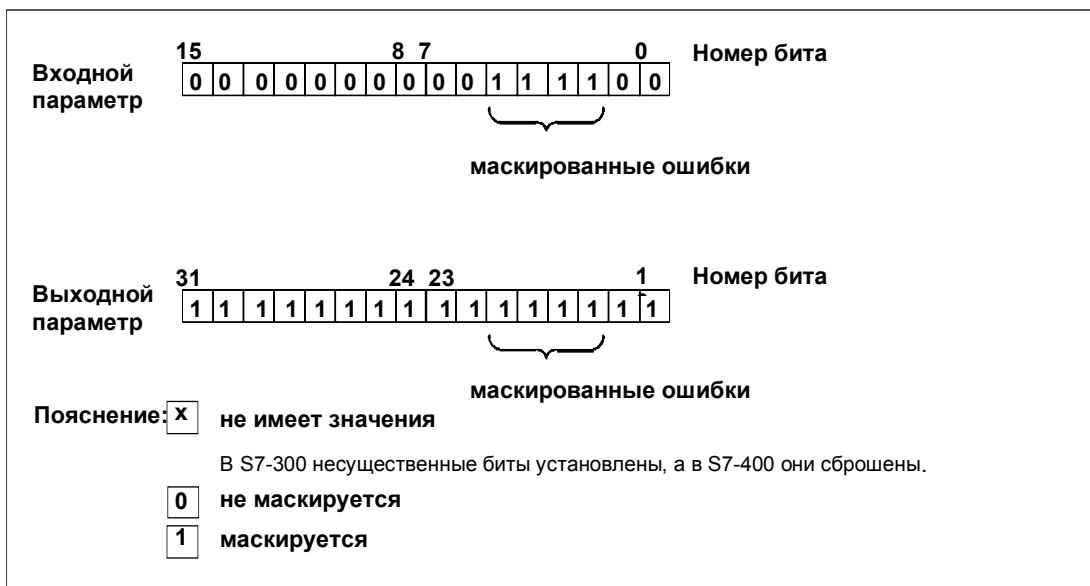
- ... входные параметры для SFC36, 37, 38 = "0"
- ... выходные параметры для SFC36, 37 = "0"
- для SFC38 = "0"



### Пример для CPU 31x (кроме CPU 318)

Следующий рисунок показывает младшее слово фильтра ошибок доступа со всеми маскированными ошибками для всех CPU, кроме CPU 417 и CPU 417H:

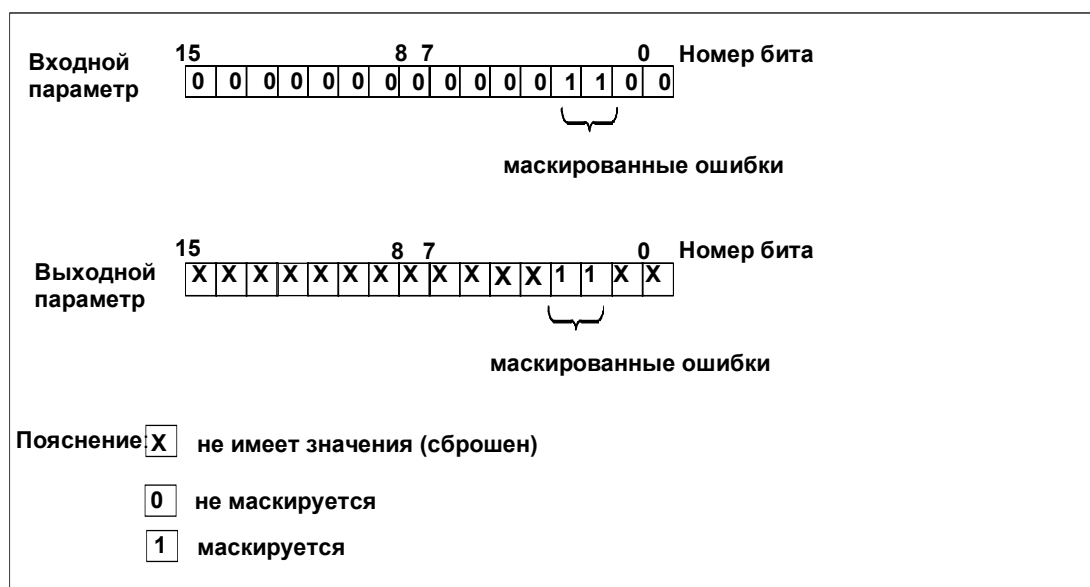
- в качестве входного параметра для SFC36
- в качестве выходного параметра для SFC36



**Пример для CPU 417 и CPU 417H**

Следующая схема показывает, как выглядит младшее слово фильтра ошибок доступа со всеми маскированными ошибками для CPU 417 и CPU 417H:

- в качестве входного параметра для SFC36
- в качестве выходного параметра для SFC36.



### Младшее слово фильтра ошибок программирования

В следующей таблице перечислены ошибки, поставленные в соответствие младшему слову фильтра ошибок программирования. Таблица показывает также возможные причины ошибок.

Возможные причины ошибок программирования, младшее слово:

Ошибка	ID события (W#16#...)	Ошибка происходит ...
Ошибка преобразования BCD	2521	... когда преобразуемое значение не является числом в двоично-десятичном коде (например, 5E8)
Ошибка длины области при чтении	2522	... когда используется адрес, который не находится полностью в пределах возможной адресной области. Пример: нужно прочитать MW 320, хотя область памяти имеет длину только 256 байтов.
Ошибка длины области при записи	2523	... когда используется адрес, который не находится полностью в пределах возможной адресной области. Пример: нужно записать значение в MW 320, хотя область памяти имеет длину только 256 байтов.
Ошибка области при чтении	2524	... когда для адреса задан неправильный идентификатор области при использовании косвенной межсегментной адресации. Пример: правильно: LAR1 P#E 12.0 L W[AR1, P#0.0] неправильно: LAR1 P#12.0 L W[AR1, P#0.0] Для этой операции выводится сообщение об ошибке длины области.
Ошибка области при записи	2525	... когда для адреса задан неправильный идентификатор области при использовании косвенной межсегментной адресации. Пример: правильно: LAR1 P#E 12.0 T W[AR1, P#0.0] неправильно: LAR1 P#12.0 T W[AR1, P#0.0] Для этой операции выводится сообщение об ошибке длины области.
Ошибка номера таймера	2526	... когда происходит обращение к несуществующему таймеру. Пример: SP T [MW 0], где MW 0 = 129; должен запускаться таймер 129, хотя имеются только 128 доступных таймеров.
Ошибка номера счетчика	2527	... когда происходит обращение к несуществующему счетчику. Пример: CU C [MW 0], где MW 0 = 600; должно происходить обращение к счетчику 600, хотя имеются только 512 доступных счетчиков (CPU 416-D).
Ошибка выравнивания при чтении	2528	... когда адрес байта, слова или двойного слова указан с адресом бита $\neq 0$ . Пример: правильно: LAR1 P#M12.0 L B[AR1, P#0.0] неправильно: LAR1 P#M12.4 L B[AR1, P#0.0]

Ошибка	ID события (W#16#...)	Ошибка происходит ...
Ошибка выравнивания при записи	2529	... когда адрес байта, слова или двойного слова указан с адресом бита $\neq 0$ . Пример:   правильно: LAR1 P#M12.0 T B[AR1, P#0.0] неправильно: LAR1 P#M12.4 T B[AR1, P#0.0]

### Старшее слово фильтра ошибок программирования

В следующей таблице перечислены ошибки, поставленные в соответствие старшему слову фильтра ошибок программирования. Перечисляются также возможные причины ошибок.

Возможные причины ошибок программирования, старшее слово:

Ошибка	ID события (W#16#...)	Ошибка происходит ...
Ошибка записи в блок данных	2530	... когда блок данных, в который должна производиться запись, доступен только для чтения.
Ошибка записи в экземпляр блок данных	2531	... когда экземпляр блок данных, в который должна производиться запись, доступен только для чтения.
Ошибка номера блока DB	2532	... когда должен открываться блок данных с номером, превышающим максимальный разрешенный номер.
Ошибка номера блока DI	2533	... когда должен открываться экземпляр блок данных с номером, превышающим максимальный разрешенный номер.
Ошибка номера блока FC	2534	... когда вызывается функция с номером, превышающим максимальный разрешенный номер.
Ошибка номера блока FB	2535	... когда вызывается функциональный блок с номером, превышающим максимальный разрешенный номер.
DB не загружен	253A	... когда открываемый блок данных не загружен.
FC не загружен	253C	... когда вызываемая функция не загружена.
SFC не существует	253D	... когда вызываемая системная функция не существует.
FB не загружен	253E	... когда вызываемый функциональный блок не загружен.
SFB не существует	253F	... когда вызываемый системный/ стандартный функциональный блок не существует.

**Ошибки доступа**

В следующей таблице перечислены ошибки, поставленные в соответствие фильтру ошибок доступа во всех CPU. Перечисляются также возможные причины ошибок.

Ошибка	ID события (W#16#...)	Ошибка происходит...
Ошибка доступа к входам/выходам при чтении	2942	... когда адресу в области входов/выходов не назначен сигнальный модуль или ... когда обращение к этой области входов/выходов не подтверждается в течение выбранного контрольного времени модуля (таймаута).
Ошибка доступа к входам/выходам при записи	2943	... когда адресу в области входов/выходов не назначен сигнальный модуль или ... когда обращение к этой области входов/выходов не подтверждается в течение выбранного контрольного времени модуля (таймаута).

## 11.2 Маскирование синхронных ошибок с помощью SFC36 "MSK\_FLT"

### Описание

С помощью SFC36 "MSK\_FLT" (mask synchronous errors [маскировать синхронные ошибки]) Вы можете управлять реакцией CPU на синхронные ошибки. С помощью этой SFC Вы можете маскировать синхронные ошибки, используя фильтр ошибок (см. предыдущий раздел). При вызове SFC36 Вы маскируете синхронные ошибки в текущем классе приоритета.

Если Вы устанавливаете в "1" отдельные биты фильтра синхронных ошибок во входных параметрах, то другие ранее установленные биты сохраняют свое значение "1". Следовательно, Вы получаете новые фильтры ошибок, которые Вы можете считывать через выходные параметры. Маскированные вами синхронные ошибки не вызывают OB, а просто вводятся в регистр ошибок. Вы можете считывать регистр ошибок с помощью SFC38 "READ\_ERR" (см. раздел "Чтение регистра ошибок с помощью SFC38 "READ\_ERR").

Параметр	Описание	Тип данных	Область памяти	Характеристика
PRGFLT_SET_MASK	INPUT	DWORD	I, Q, M, D, L, константа	Ошибки программирования, подлежащие маскированию.
ACCFLT_SET_MASK	INPUT	DWORD	I, Q, M, D, L, константа	ошибки доступа, подлежащие маскированию
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках
PRGFLT_MASKED	OUTPUT	DWORD	I, Q, M, D, L	Маскированные ошибки программирования
ACCFLT_MASKED	OUTPUT	DWORD	I, Q, M, D, L	Маскированные ошибки доступа

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Никакая из ошибок еще не была маскирована.
0001	По крайней мере, одна из ошибок была уже маскирована. Тем не менее, другие ошибки будут маскироваться.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

### 11.3 Демаскирование синхронных ошибок с помощью SFC37 "DMSK\_FLT"

#### Описание

С помощью SFC37 "DMSK\_FLT" (unmask synchronous errors [демаскировать синхронные ошибки]) демаскируются синхронные ошибки, маскированные с помощью SFC36 "MSK\_FLT". Для этого Вы должны установить в "1" соответствующие биты фильтра ошибок во входных параметрах (см. раздел 10.1). Посредством вызова SFC 37 Вы демаскируете соответствующие синхронные ошибки текущего класса приоритета. Одновременно стираются записи в регистре ошибок. Вы можете считывать новые фильтры ошибок через выходные параметры.

Параметр	Описание	Тип данных	Область памяти	Характеристика
PRGFLT_RESET_MASK	INPUT	DWORD	I, Q, M, D, L, константа	Ошибки программирования, подлежащие демаскированию
ACCFLT_RESET_MASK	INPUT	DWORD	I, Q, M, D, L, константа	Ошибки доступа, подлежащие демаскированию
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках
PRGFLT_MASKED	OUTPUT	DWORD	I, Q, M, D, L	Все еще маскированные ошибки программирования
ACCFLT_MASKED	OUTPUT	DWORD	I, Q, M, D, L	Все еще маскированные ошибки доступа

#### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Все указанные ошибки были демаскированы.
0001	По крайней мере, одна из ошибок не была маскирована. Тем не менее, другие ошибки будут демаскированы.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 11.4 Чтение регистра ошибок с помощью SFC38 "READ\_ERR"

### Описание

С помощью SFC38 "READ\_ERR" (read error register [прочитать регистр ошибок]) Вы можете считать регистр ошибок. Структура регистра ошибок соответствует структуре фильтров ошибок программирования и доступа, которые Вы можете запрограммировать в качестве входных параметров с помощью SFC36 и SFC37.

Во входных параметрах Вы вводите синхронные ошибки, которые Вы хотите считывать из регистра ошибок. Когда Вы вызываете SFC38, Вы считываете желаемые записи из регистра ошибок и одновременно стираете эти записи.

Регистр ошибок содержит информацию о том, какие из маскированных синхронных ошибок в текущем классе приоритета произошли хотя бы один раз. Если бит установлен, значит, соответствующая маскированная синхронная ошибка появились, по крайней мере, один раз.

Параметр	Описание	Тип данных	Область памяти	Характеристика
PRGFLT_QUERY	INPUT	DWORD	I, Q, M, D, L, константа	Опрос ошибок программирования
ACCFLT_QUERY	INPUT	DWORD	I, Q, M, D, L, константа	Опрос ошибок доступа
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках
PRGFLT_ESR	OUTPUT	DWORD	I, Q, M, D, L	Появившаяся ошибка программирования
ACCFLT_ESR	OUTPUT	DWORD	I, Q, M, D, L	Появившаяся ошибка доступа

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Все опрошенные ошибки маскированы.
0001	По крайней мере одна из опрошенных ошибок не маскирована.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL



## **12 SFC для обработки прерываний и асинхронных ошибок**

## 12.1 Задержка и блокировка прерываний и асинхронных ошибок

### Назначение SFC39 – SFC42

С помощью этих SFC Вы можете достичь следующего:

- Заблокировать прерывания и асинхронные ошибки с помощью SFC39 "DIS\_IRT" на время всех последующих циклов CPU.
- Задержать более высокие классы приоритета с помощью SFC41 "DIS\_AIRT" до конца OB.
- Разблокировать прерывания и асинхронные ошибки с помощью SFC40 "EN\_IRT" или SFC42 "EN\_AIRT".

Вы программируете обработку прерываний и асинхронных ошибок в программе пользователя. Для этого Вы должны запрограммировать также соответствующие OB.

### Преимущество SFC41 и SFC42

Задержка высокоприоритетных прерываний и асинхронных ошибок путем их блокировки с помощью SFC 41 "DIS\_AIRT" и последующей разблокировки с помощью SFC 42 "EN\_AIRT" имеет следующие преимущества:

Количество задержанных прерываний подсчитывается CPU. Если Вы задержали прерывания и асинхронные ошибки, то эта задержка не может быть отменена стандартными вызовами FC, если эти прерывания и асинхронные ошибки также блокируются и вновь разблокируются в самих этих стандартных FC.

### Классы прерываний

Прерывания подразделены на разные классы. Следующая таблица перечисляет все классы прерываний и соответствующие OB.

Класс прерывания	OB
Прерывания по времени	OB10 ... OB17
Прерывания с задержкой	OB20 ... OB23
Циклические прерывания	OB30 ... OB38
Аппаратные прерывания	OB40 ... OB47
Прерывания для DPV1	OB55 ... OB57
Прерывания многопроцессорной обработки	OB60
Прерывания ошибок резервирования	OB70, OB72
Прерывания асинхронных ошибок	OB80 ... OB87 (см. ниже)
Прерывания синхронных ошибок	OB121, OB122 (Вы можете маскировать или демаскировать обработку прерываний синхронных ошибок с помощью SFC36 ... SFC38)

### Асинхронные ошибки

Следующая таблица перечисляет все асинхронные ошибки, на которые Вы можете реагировать с помощью вызова ОВ в программе пользователя.

Асинхронные ошибки	ОВ
Ошибка времени (например, превышение времени цикла)	ОВ80
Неисправность источника питания (например, отказ батарейки)	ОВ81
Диагностическое прерывание (например, неисправный предохранитель в сигнальном модуле)	ОВ82
Прерывание из-за удаления/вставки модуля	ОВ83
Аппаратная ошибка CPU (например, удалена плата памяти)	ОВ84
Ошибка исполнения программы	ОВ85
Выход из строя стойки	ОВ86
Коммуникационная ошибка	ОВ87

## 12.2 Блокировка обработки новых прерываний и асинхронных ошибок с помощью SFC39 "DIS\_IRT"

### Описание

С помощью SFC 39 "DIS\_IRT" (disable interrupt [заблокировать прерывание]) Вы блокируете обработку новых прерываний и асинхронных ошибок. Это означает, что при возникновении прерывания операционная система CPU реагирует следующим образом:

- Она **не вызывает** ни ОБ прерываний, ни ОБ асинхронных ошибок и
- **не запускает** стандартную реакцию, если ОБ прерываний или ОБ асинхронных ошибок не запрограммирован.

Блокировка прерываний и асинхронных ошибок остается в силе для всех классов приоритета. Воздействие "DIS\_IRT" может быть отменено только вызовом SFC40 "EN\_IRT" (см. раздел 0) или теплым или холодным рестартом.

Записывает или нет операционная система в диагностический буфер прерывания и асинхронные ошибки при их возникновении, зависит от выбранной Вами настройки входного параметра MODE [режим].

### Примечание

Помните, что при программировании использования SFC39 "DIS\_IRT" все происходящие прерывания теряются!

Параметр	Описание	Тип данных	Область памяти	Характеристика
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Указывает, какие прерывания и асинхронные ошибки блокируются (см. ниже таблицу MODE).
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер ОБ
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит, когда эта функция активна, то возвращаемое значение содержит код ошибки.

**MODE**

<b>MODE</b>	<b>Значение</b>
00	Вновь появляющиеся прерывания и асинхронные ошибки блокируются. (синхронные ошибки не блокируются). Назначьте параметру OB_NR значение 0. Записи в диагностический буфер продолжают заноситься.
01	Все вновь появляющиеся события заданного класса прерываний блокируются. Класс прерываний задается указанием младшего (первого) по номеру OB для данного класса, даже если данный блок не существует в Вашем CPU. Примеры: - Если необходимо блокировать прерывания по времени (watchdog), задайте в параметре OB_NR значение 30 (даже если OB30 не является первым в этом классе для Вашего CPU). - Если необходимо блокировать все аппаратные прерывания, задайте в параметре OB_NR значение 40. Записи в диагностический буфер будут продолжать заноситься.
01	Все вновь появляющиеся события заданного класса прерываний блокируются. Класс прерываний задается в соответствии с правилом: - Прерывания по времени: 10 - Прерывания с задержкой: 20 - Циклические прерывания: 30 - Аппаратные прерывания: 40 - Прерывания для DPV1: 50 - Прерывания многопроцессорной обработки: 60 - Прерывания ошибок резервирования: 70 - Прерывания асинхронных ошибок: 80
02	Все новые появления заданного прерывания блокируются. Это прерывание задается с помощью номера OB. Записи в диагностический буфер продолжают заноситься.
80	Все вновь появляющиеся прерывания и асинхронные ошибки блокируются и более не вносятся в диагностический буфер. Операционная система вносит в диагностический буфер событие W#16#5380.
81	Все вновь появляющиеся прерывания, принадлежащие заданному классу прерываний, блокируются и более не вносятся в диагностический буфер. Операционная система вносит в диагностический буфер событие W#16#5380.
82	Все вновь появляющиеся события, относящиеся к заданному прерыванию, блокируются и более не вносятся в диагностический буфер. Операционная система вносит в диагностический буфер событие W#16#5380.

**Информация об ошибках**

<b>Код ошибки (W#16#...)</b>	<b>Описание</b>
0000	Ошибки не было.
8090	Входной параметр OB_NR содержит недопустимое значение.
8091	Входной параметр MODE содержит недопустимое значение.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 12.3 Разблокировка обработки новых прерываний и асинхронных ошибок с помощью SFC40 "EN\_IRT"

### Описание

С помощью SFC 40 "EN\_IRT"(enable interrupt [разблокировать прерывание]) Вы разблокируете обработку новых прерываний и асинхронных ошибок, заблокированную посредством SFC39 "DIS\_IRT". Это значит, что при возникновении события прерывания операционная система CPU реагирует одним из следующих способов:

- Вызывает ОВ прерываний или ОВ асинхронных ошибок.
- Запускает стандартную реакцию, если нет запрограммированного ОВ прерываний или ОВ асинхронных ошибок.

Параметр	Описание	Тип данных	Область памяти	Характеристика
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Указывает, какие прерывания и асинхронные ошибки будут разблокированы.
OB_NR	INPUT	INT	I, Q, M, D, L, константа	Номер ОВ
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит, когда эта функция активна, то возвращаемое значение содержит код ошибки.

### MODE

MODE	Значение
0	Все вновь появляющиеся прерывания и асинхронные ошибки разблокируются.
1	Все вновь появляющиеся события заданного класса прерываний блокируются. Класс прерываний задается в соответствии с правилом: - Прерывания по времени: 10 - Прерывания с задержкой: 20 - Циклические прерывания: 30 - Аппаратные прерывания: 40 - Прерывания для DPV1: 50 - Прерывания многопроцессорной обработки: 60 - Прерывания ошибок резервирования: 70 - Прерывания асинхронных ошибок: 80
2	Все вновь появляющиеся события заданного прерывания разблокируются. Это прерывание задается номером ОВ.

### Информация об ошибках

Код ошибки (W#16#...)	Описание
0000	Ошибки не было.
8090	Входной параметр OB_NR содержит недопустимое значение.
8091	Входной параметр MODE содержит недопустимое значение.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 12.4 Задержка обработки прерываний более высокого приоритета и асинхронных ошибок с помощью SFC41 "DIS\_AIRT"

### Описание

С помощью SFC 41 "DIS\_AIRT" (disable alarm interrupts [заблокировать аварийное прерывание]) Вы задерживаете обработку ОВ прерываний и ОВ асинхронных ошибок, приоритет которых выше, чем приоритет текущего ОВ. Вы можете вызывать SFC 41 в ОВ многократно. Вызовы SFC 41 подсчитываются операционной системой.

Каждый из этих вызовов остается в силе, пока он не будет вновь отменен индивидуально вызовом SFC42 "EN\_AIRT" или пока текущий ОВ не будет полностью обработан.

Прерывания и асинхронные ошибки, которые произошли во время действия SFC41, обрабатываются, как только они вновь разблокируются с помощью SFC42 "EN\_AIRT" или как только будет исполнен текущий ОВ.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Количество задержек (= количеству вызовов SFC 41)

### Возвращаемое значение

Следующая таблица показывает возвращаемое значение для SFC 41, которое выводится через параметр RET\_VAL.

Возвращаемое значение	Описание
n	"n" показывает, сколько раз обработка была заблокирована, иными словами, число вызовов SFC41 (обработка прерываний опять разблокируется лишь тогда, когда n = 0; см. раздел "Разблокировка обработки прерываний более высокого приоритета и асинхронных ошибок с помощью SFC42 "EN_AIRT").

## 12.5 Разблокировка обработки прерываний более высокого приоритета и асинхронных ошибок с помощью SFC42 "EN\_AIRT"

### Описание

С помощью SFC 42 "EN\_AIRT" (enable alarm interrupts [разблокировать аварийные прерывания]) Вы разблокируете SFC 41 обработку прерываний более высокого приоритета и асинхронных ошибок, заблокированную ранее с помощью "DIS\_AIRT". Каждый вызов SFC41 должен быть отменен вызовом SFC42.

### Пример

Если Вы, например, пять раз заблокировали прерывания с помощью пяти вызовов SFC41, то Вы должны отменить эти вызовы пятью вызовами SFC42.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Количество все еще запрограммированных задержек после выполнения SFC 42 или сообщение об ошибке.

### Возвращаемое значение и информация об ошибках

Обратитесь к материалу по проверке на предмет ошибок с помощью выходного параметра RET\_VAL.

Как оценивать информацию об ошибках из параметра RET\_VAL, объяснено в главе 0. В этой главе Вы также найдете общую информацию по ошибкам для SFC. Следующая таблица содержит всю информацию об ошибках, относящуюся к SFC42, которая может быть выведена с помощью параметра RET\_VAL.

Возвращаемое значение и информация об ошибках	Описание
n	"n" показывает количество вызовов SFC41, еще не отмененных вызовами SFC42 (обработка прерываний вновь разблокируется только при "n" = 0).
W#16#8080	Функция была вызвана вновь, хотя обработка прерываний уже была разблокирована.



## **13 SFC для диагностики**

## 13.1 Системная диагностика

CPU поддерживают внутренние данные о состоянии программируемого логического контроллера. С помощью системных диагностических функций Вы можете считывать наиболее важные данные. Некоторые из этих данных можно отображать на устройстве программирования, используя STEP 7. Вы можете также обращаться к данным, требуемым для диагностики системы в своей программе, используя SFC "RD\_SINFO" и "RDSYSST".

## 13.2 Чтение стартовой информации ОБ с помощью SFC6 "RD\_SINFO"

### Описание

С помощью SFC6 "RD\_SINFO" (read start information [читать стартовую информацию]) Вы можете прочитать стартовую информацию, содержащую следующие сведения:

- Последний вызванный ОБ, который еще не полностью выполнен и
- Последний ОБ запуска, который должен быть запущен.

В обоих случаях метка времени отсутствует. Если вызов происходит в ОБ100 ОБ101 или ОБ102, то возвращаются два идентичных сообщения со стартовой информацией.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке
TOP_SI	OUTPUT	STRUCT	D, L	Стартовая информация текущего ОБ
START_UP_SI	OUTPUT	STRUCT	D, L	Стартовая информация запущенного последним ОБ запуска

### TOP\_SI и START\_UP\_SI

Выходные параметры TOP\_SI и START\_UP\_SI являются двумя структурами с одинаковыми элементами (см. следующую таблицу).

Элемент структуры	Тип данных	Описание
EV_CLASS	BYTE	<ul style="list-style-type: none"> <li>• Биты с 0 по 3: Идентификатор события</li> <li>• Биты с 4 по 7: Класс события</li> </ul>
EV_NUM	BYTE	Номер события
PRIORITY	BYTE	Номер класса приоритета
NUM	BYTE	Номер ОБ
TYP2_3	BYTE	Идентификатор данных 2_3: характеризует информацию, записанную в ZI2_3
TYP1	BYTE	Идентификатор данных 1: характеризует информацию, записанную в ZI1
ZI1	WORD	Дополнительная информация 1
ZI2_3	DWORD	Дополнительная информация 2_3

### Примечание

Элементы структуры, перечисленные в таблице и временные переменные ОБ имеют идентичное содержимое.

Обратите внимание на то, что временные переменные отдельных ОБ могут иметь различные имена и различные типы данных. Обратите также внимание на то, что интерфейс вызова каждого ОБ включает в себя дополнительную информацию, содержащую дату и время запроса ОБ.

Биты с 4 по 7 элемента структуры EV\_CLASS содержат класс события. Здесь возможны следующие значения:

- 1: Стартовые события из стандартных ОБ
- 2: Стартовые события из ОБ синхронных ошибок
- 3: Стартовые события из ОБ асинхронных ошибок

Структурный элемент PRIORITY передает класс приоритета, относящийся к текущему ОБ (см. главу 0).

Кроме этих двух элементов, имеет значение также параметр NUM. NUM содержит номер текущего ОБ или запущенного последним ОБ запуска.

### Пример

Пусть последним был вызван и еще не полностью обработан ОБ 80. Пусть последним запущенным ОБ запуска будет ОБ 100.

Следующая таблица показывает соответствие между структурными элементами параметра TOP\_SI функции SFC 6 "RD\_SINFO" и локальными переменными ОБ 80.

TOP_SI		ОБ 80	
Элемент структуры	Тип данных	Локальная переменная	Тип данных
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

Следующая таблица показывает соответствие между структурными элементами параметра START\_UP\_SI функции SFC 6 "RD\_SINFO" и локальными переменными OB 100.

START_UP_SI		OB 100	
Элемент структуры	Тип данных	Локальная переменная	Тип данных
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

#### Информация об ошибках

SFC6 "RD\_SINFO" предоставляет только общую, а не специфическую информацию об ошибках. Коды общих ошибок и способы их оценки описаны в разделе "Общие параметры для SFC".

### 13.3 Чтение списка состояний системы или подписка с помощью SFC51 "RDSYSST"

#### Описание

С помощью системной функции SFC51 "RDSYSST" (read system status [читать состояние системы]) читают список состояний системы или частичный список состояний системы.

Чтение запускают, присваивая при вызове SFC51 входному параметру REQ значение 1. Если состояние системы можно прочитать сразу, то SFC возвращает в выходном параметре BUSY значение 0. Если BUSY имеет значение 1, то функция чтения еще не завершилась (см. раздел 0).

#### Примечание

Если Вы вызываете SFC51 "RDSYSST" в ОВ диагностического прерывания с SSL-ID W#16#00B1 или W#16#00B2 или W#16#00B3 и обращаетесь к модулю, который инициировал диагностическое прерывание, то состояние системы читается немедленно.

#### Системные ресурсы

Если Вы запускаете несколько асинхронных функций чтения (задания с SSL\_ID W#16#00B4, и W#16#4C91, и W#16#4092, и W#16#4292, и W#16#4692, и, возможно, W#16#00B1, и W#16#00B3) одну за другой через короткие интервалы времени, то операционная система гарантирует, что все задания на чтение выполняются и что они не создают помех друг для друга. Если достигаются пределы системных ресурсов, то это отображается в RET\_VAL. Вы можете исправить эту ситуацию нерегулярной ошибки, повторив задание.

Максимальное число "одновременно" активированных заданий SFC51 зависит от CPU. Вы найдете эту информацию в [/70/](#) и [/101/](#).

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ=1: запускает обработку.
SSL_ID	INPUT	WORD	I, Q, M, D, L, константа	Идентификатор списка состояний системы или частичного списка, который нужно читать (частичные списки объясняются в главе 27).
INDEX	INPUT	WORD	I, Q, M, D, L, константа	Тип или номер объекта в частичном списке.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при выполнении SFC происходит ошибка, то параметр RET_VAL содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	TRUE: Чтение еще не завершилось.
SSL_HEADER	OUTPUT	STRUCT	D, L	См. ниже.
DR	OUTPUT	ANY	I, Q, M, L, D	Целевая область для прочитанного списка SSL или для прочитанного частичного списка SSL: <ul style="list-style-type: none"> <li>Если Вы считали информацию только заголовка списка SSL, то Вы должны оценивать не DR, а только SSL_HEADER.</li> <li>В противном случае, произведение LENGTHDR и N_DR показывает, сколько байтов были введены в DR.</li> </ul>

**SSL\_HEADER**

Параметр SSL\_HEADER является структурой, определенной следующим образом:

```
SSL_HEADER: STRUCT
  LENGTHDR: WORD
  N_DR:     WORD
END_STRUCT
```

LENGTHDR - это длина записи данных списка SSL или частичного списка SSL.

- Если Вы считали информацию только заголовка списка SSL, то N\_DR содержит количество принадлежащих ему записей данных.
- В противном случае N\_DR содержит количество записей данных, переданных в область назначения.

**Информация об ошибках**

Код ошибки (W#16#...)	Характеристика
0000	Ошибок нет.
0081	Поле результата слишком короткое. (Тем не менее, передается столько записей данных, сколько возможно. Заголовок SLZ указывает это количество).
7000	Первый вызов с REQ=0: Передача данных не активна; BUSY имеет значение 0.
7001	Первый вызов с REQ=1: Передача данных запущена; BUSY имеет значение 1.
7002	Промежуточный вызов (REQ не имеет значения): Передача данных уже активна; BUSY имеет значение 1.
8081	Поле результата слишком короткое (недостаточно места для одной записи данных).
8082	SSL_ID неверен или неизвестен в CPU или SFC.
8083	Неправильный или неразрешенный INDEX.
8085	Из-за проблемы в системе информация в настоящее время недоступна (например, из-за недостатка ресурсов).
8086	Запись данных не может быть прочитана из-за ошибки системы (шина, модули, операционная система).
8087	Запись данных не может быть прочитана, потому что модуль не существует или не подтверждает.
8088	Запись данных не может быть прочитана, потому что фактический идентификатор типа отличается от ожидаемого идентификатора типа.
8089	Запись данных не может быть прочитана, потому что модуль не обладает диагностическими свойствами.
80A2	Ошибка протокола DP (ошибка уровня 2) (нерегулярная ошибка).
80A3	Ошибка протокола DP у пользовательского интерфейса / пользователя (нерегулярная ошибка).
80A4	Ошибка связи в коммуникационной шине (ошибка происходит между CPU и внешним интерфейсным модулем DP).
80C5	Децентрализованная периферия недоступна (нерегулярная ошибка).
80C6	Передача записи данных прекратилась из-за прерывания класса приоритета (рестарт или фоновый режим).
80D2	Запись данных не может быть считана, т.к. модуль не имеет средств диагностики.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## SSL\_ID

**Примечание**

По поводу отдельных списков, которые можно считывать с помощью SFC51 "RDSYSST", обратитесь

- к /70/ для S7-300
- к следующей таблице для S7-400.

SSL_ID (W#16#...)	Частичный список	INDEX (W#16#...)
	<b>Идентификатор (ID) модуля</b>	
0111	Одна идентифицирующая запись данных	
	Идентификация модуля	0001
	Идентификация основного комплекта оборудования	0006
	Идентификация основного комплекта оборудования	0007
	<b>Характеристики CPU</b>	
0012	Все характеристики	Не имеет значения
0112	Характеристики одной группы	
	Процессор MC7	0000
	Система времени	0100
	Поведение системы	0200
	Описание языка MC7	0300
0F12	Информация только заголовка частичного списка SSL	Не имеет значения
	<b>Области памяти пользователя</b>	
0113	Одна запись данных для указанной области памяти	
	Рабочая память	0001
	<b>Системные области</b>	
0014	Записи данных всех системных областей	Не имеет значения
0F14	Информация только заголовка частичного списка SSL	Не имеет значения
	<b>Типы модулей</b>	
0015	Записи данных всех типов модулей	Не имеет значения
	<b>Состояние светодиодов модулей</b> (не может считываться из всех CPU, см. /102 /).	
0019	Состояние всех светодиодов	Не имеет значения
0F19	Информация только заголовка частичного списка SSL	Не имеет значения
	<b>Идентификация одного компонента</b>	
001C	Идентификация всех компонентов	Не имеет значения
011C	Идентификация одного компонента	
	Наименование системы автоматического управления	0001
	Наименование CPU	0002
	Системный идентификатор (ID) CPU	0003



SSL_ID (W#16#...)	Частичный список	INDEX (W#16#...)
	Ввод копирайта	0004
	зарезервирован для операционной системы	0006
0F1C	Информация заголовка только для SSL списка	Не имеет значения
	<b>Состояние прерывания</b>	
0222	Запись данных указанного прерывания	Номер OB
	<b>Данные о состоянии связи</b>	
0132	Данные о состоянии одного устройства связи	
	Диагностика	0005
	Система времени	0008
0232	Данные о состоянии одного устройства связи	
	Уровень защиты CPU и параметры настройки операторского управления	
	<b>Групповая информация H CPU</b>	
0071	Информация о текущем состоянии H-системы	Не имеет значения
0F71	Информация только заголовка частичного списка SSL	Не имеет значения
	<b>Состояние светодиодов модулей</b> (не может считываться из всех CPU, см. /102 /).	
0174	Состояние светодиода	LED ID

SSL_ID (W#16#...)	Частичный список	INDEX (W#16#...)
	Системная информация ведущего DP-устройства	
0090	Информация о системах ведущих DP-устройств, которые известны CPU	0000
0190	Информация о системе ведущего DP-устройства	ID системы ведущего DP-устройства
0F90	Информация только заголовка частичного списка SSL	0000
	<b>Информация о состоянии модуля</b> (передается не более 27 записей данных)	
0091	Информация о состоянии всех вставленных модулей / submodule	Не имеет значения
0191	Информация о состоянии всех модулей / стоек с неправильным идентификатором типа	Не имеет значения
0291	Информация о состоянии всех неисправных модулей	Не имеет значения
0391	Информация о состоянии всех недоступных модулей	Не имеет значения
0591	Информация о состоянии всех submodule главного модуля	Не имеет значения
0991	Информация о состоянии всех submodule главного модуля в указанной стойке	Идентификатор стойки или master-системы DP
0A91	Информация о состоянии модулей всех master-систем DP	Не имеет значения
0C91	Информация о состоянии модуля в центральной конфигурации или модуля, подключенного к встроенному коммуникационному процессору DP	Логический базовый адрес
4C91	Информация о состоянии модуля, подключенного к внешнему коммуникационному процессору DP	Логический базовый адрес

SSL_ID (W#16#...)	Частичный список	INDEX (W#16#...)
0D91	Информация о состоянии всех модулей в указанной стойке / станции DP	Идентификатор (ID) стойки или master-системы DP или ID master-системы DP и номер станции
0E91	Информация о состоянии всех размещенных модулей	Не имеет значения
	<b>Информация о состоянии стойки / станции</b>	
0092	Ожидаемое состояние стойки в центральной конфигурации / станций master-системы DP	0 / ID master-системы DP
4092	Ожидаемое состояние станций master-системы DP, подключенной к внешнему интерфейсу DP	ID master-системы DP
0292	Фактическое состояние стойки в центральной конфигурации / станций master-системы DP	0 / ID master-системы DP
4292	Фактическое состояние станций master-системы DP, подключенной через внешний интерфейсный модуль DP	ID master-системы DP
0692	Нормальное состояние стоек расширения в центральной конфигурации / станций master-системы DP, подключенной через встроенный интерфейсный модуль DP	0 / ID master-системы DP
4692	Нормальное состояние станций master-системы DP, подключенной через внешний интерфейсный модуль DP	ID master-системы DP
	<b>Диагностический буфер</b> (максимум 21 запись данных)	
00A0	Все записи, которые могут быть поставлены в активном режиме работы в настоящее время	Не имеет значения
01A0	Самые недавние записи, количество задается в индексе	Количество
0FA0	Информация только заголовка частичного списка SSL	Не имеет значения
	<b>Диагностические данные модулей</b>	
00B1	Первые четыре диагностических байта одного модуля (запись данных 0)	Логический базовый адрес
00B2	Все диагностические данные одного модуля (□220 байтов, запись данных 1) (не модули DP)	Стойка, слот
00B3	Все диагностические данные одного модуля (□220 байтов, запись данных 1)	Логический базовый адрес
00B4	Диагностические данные ведомые DP-устройства	Конфигурированный диагностический адрес

## 13.4 Запись диагностического события, определенного пользователем, в диагностический буфер с помощью SFC52 "WR\_USMSG"

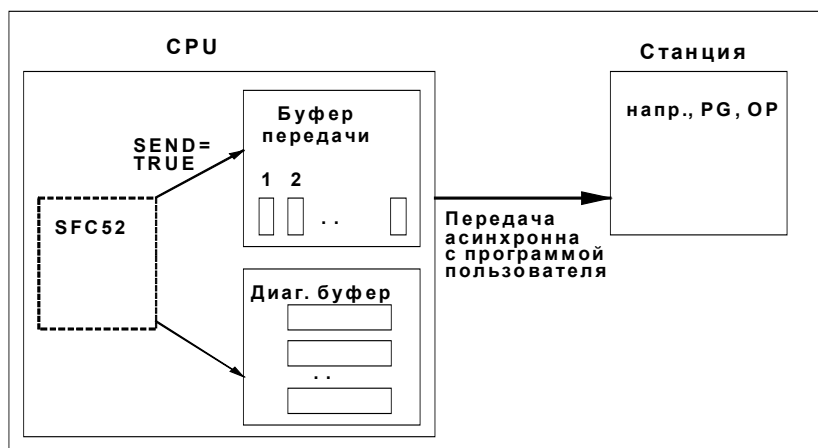
### Описание

С помощью SFC 52 "WR\_USMSG" (write user element in diagnosis buffer [записать элемент пользователя в диагностический буфер]) определенное пользователем диагностическое событие записывается в диагностический буфер. Кроме того, Вы можете передать соответствующее диагностическое сообщение всем зарегистрированным для этой цели станциям. Если появится ошибка, то выходной параметр RET\_VAL предоставляет информацию об ошибке.

### Передача определенного пользователем диагностического сообщения

Определенное пользователем диагностическое событие записывается в буфер диагностики с помощью SFC 52. Вы можете также передать соответствующее диагностическое сообщение всем зарегистрированным для этой цели станциям. Тогда определенное пользователем диагностическое сообщение записывается в буфер передачи и оттуда автоматически передается зарегистрированным для этого станциям.

Вы можете проверить, возможна ли в данный момент передача определенных пользователем диагностических сообщений. Для этого вызовите SFC 51 "RDSYSST" с параметрами SSL\_ID = W#16#0132 и INDEX = W#16#0005. Четвертое слово полученной записи данных показывает, возможна в данный момент передача (1) или нет (0).



### Переполнение передающего буфера

Запись диагностического сообщения в буфер передачи может происходить только тогда, когда буфер передачи неполон. Количество записей, которое может быть сделано в буфере передачи, зависит от типа используемого вами CPU.

Если буфер передачи полон, то:

- диагностическое событие вносится, тем не менее, в диагностический буфер,
- в параметре RET\_VAL указывается, что буфер передачи заполнен (RET\_VAL = W#16#8092).

### Станция не зарегистрирована

Если должно передаваться определенное пользователем диагностическое сообщение (SEND = TRUE) и ни одна станция не зарегистрирована, то

- определенное пользователем диагностическое событие вносится в диагностический буфер,
- параметр RET\_VAL указывает, что нет зарегистрированных станций (RET\_VAL = W#16#8091 или W#16#8091. Значение W#16#8091 появляется у старых версий CPU).

### Общая структура

Элемент в диагностическом буфере имеет следующую структуру:

Байт	Содержимое
1 и 2	Идентификатор (ID) события
3	Класс приоритета
4	Номер OB
5 и 6	Резерв
7 и 8	Дополнительная информация 1
9, 10, 11 и 12	Дополнительная информация 2
13 – 20	Отметка времени

### Идентификатор (ID) события

Каждому событию поставлен в соответствие ID события.

### Дополнительная информация

Это дополнительная информация о событии. Дополнительная информация может быть различна для каждого события. Когда Вы создаете диагностическое событие, то Вы можете сами определить содержание этих записей.

Когда Вы посылаете определенное пользователем диагностическое сообщение, Вы можете встроить дополнительную информацию в текст сообщения (относящийся к идентификатору события) в качестве сопутствующего значения.

## Отметка времени

Отметка времени имеет тип Date\_and\_Time.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SEND	INPUT	BOOL	I, Q, M, D, L, константа	Разрешить передачу определенного пользователем диагностического сообщения всем зарегистрированным станциям
EVENTN	INPUT	WORD	I, Q, M, D, L, константа	Идентификатор (ID) события – Его назначаете Вы. Он не назначается сервером сообщений.
INFO1	INPUT	ANY	I, Q, M, D, L	Дополнительная информация длиной в 1 слово
INFO2	INPUT	ANY	I, Q, M, D, L	Дополнительная информация длиной в 2 слова
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках

### SEND

Если SEND = TRUE, определенное пользователем диагностическое сообщение передается всем зарегистрированным станциям. Сообщение передается только тогда, когда станция зарегистрирована, и если буфер передачи не полон. Передача элемента асинхронна по отношению к программе пользователя.

### EVENTN

Параметр EVENTN содержит идентификатор (ID) определенного пользователем события. Вы можете вводить идентификаторы событий вида W#16#8xyz, W#16#9xyz, W#16#Axyz, W#16#Bxyz.

Идентификаторы в форматах W#16#8xyz и W#16#9xyz принадлежат заранее определенным событиям, идентификаторы в форматах W#16#Axyz и W#16#Bxyz принадлежат свободно определяемым событиям.

Поступающее событие отмечается с помощью x = 1, уходящее событие – с помощью x = 0. У событий в классе A и B "yz" – это номер, назначенный сообщению в конфигурации сообщений, в шестнадцатеричном формате.

Структура ID события объяснена в разделе 26.1.

### INFO1

Параметр INFO1 содержит информацию длиной в одно слово. Для INFO1 допустимы следующие типы данных:

- WORD
- INT
- ARRAY [0 to 1] OF CHAR

Вы можете встроить параметр INFO1 в текст сообщения как сопутствующее значение и, таким образом, добавить к сообщению новейшую информацию.

## INFO2

Параметр INFO2 содержит информацию длиной в два слова. Для INFO2 допустимы следующие типы данных:

- DWORD
- DINT
- REAL
- TIME
- ARRAY [0 to 3] OF CHAR

Вы можете встроить параметр INFO2 в текст сообщения как сопутствующее значение и, таким образом, добавить к сообщению новейшую информацию.

**Информация об ошибках**

<b>Код ошибки (W#16#...)</b>	<b>Объяснение</b>
0000	Нет ошибки
0091	Нет зарегистрированных станций (в диагностический буфер вносится диагностическое событие)
8083	Недопустимый тип данных INFO1
8084	Недопустимый тип данных INFO2
8085	EVENTN недопустим
8086	Недопустимая длина INFO1
8087	Недопустимая длина INFO2
8091	(Этот код ошибки появляется только у старых версий CPU). Нет зарегистрированных станций (в диагностический буфер вносится диагностическое событие).
8092	Передача в данный момент невозможна, буфер передачи полон (в диагностический буфер вносится диагностическое событие).
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 13.5 Определение времени выполнения программы OB с помощью SFC78 "OB\_RT"

### Описание

С помощью SFC78 "OB\_RT" Вы можете определять время выполнения (runtime) для отдельных OB в разные периоды времени.

### Примечание

SFC78 возвращает последние записанные значения времени для определенных OB, независимо от того, загружен этот OB или нет в настоящее время. Данные SFC78 не уничтожаются и не перезаписываются, но сбрасываются после перезапуска ("теплый" /"warm"/ запуск).

### Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
OB_NR	INPUT	INT	I, Q, M, D, L	Номер OB, последние значения времени которого запрашиваются. Допустимые номера - это все номера OB, сконфигурированные для Вашего CPU, кроме OB 121 и OB 122. Время, затраченное на обработку синхронных ошибок, включается в общее время, требуемое для обработки OB, в котором возникла ошибка. Задание номера блоков OB 121 и 122, а также блоков, которые не сконфигурированы для CPU, приведет к сообщению об ошибке. Если OB_NR=0, то будут пересылаться данные OB, который вызвал SFC. Когда SFC78 вызывается в OB 121 или в OB 122 и OB_NR=0, то выводятся все значения времени запуска OB, включая значения времени для OB 12x.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если при обработке этой функции происходит ошибка, то возвращаемое значение содержит код ошибки. Иначе, RET_VAL содержит номер OB, для которого эти данные запрошены.
PRIO	OUTPUT	INT	I, Q, M, D, L	Приоритетный класс OB выводится в параметре PRIO



Параметр	Описание	Тип данных	Область памяти	Характеристика
LAST_RT	OUTPUT	DINT	I, Q, M, D, L	<p>Время выполнения (в мс) для самого последнего выполнения заданного ОВ. Если в текущий момент обрабатывается ОВ, для которого запрошено время выполнения, то после первого вызова SFC78 во время выполнения нужного ОВ LAST_RT выводит последнее общее время выполнения ОВ.</p> <p>С каждым последующим вызовом SFC78 во время выполнения нужного ОВ, параметр LAST_RT</p> <ul style="list-style-type: none"> <li>показывает DW#16#FFFF FFFF, если требуемый ОВ уже вызвал SFC78 (если ОВ_NR=0).</li> <li>показывает последнее время выполнения ОВ, если этот ОВ не вызывал SFC78 с ОВ_NR=0.</li> </ul> <p>Примечание: Время обработки прерываний, вызванных блоками ОВ с более высоким приоритетом, не включаются в значения LAST_RT.</p>
LAST_ET	OUTPUT	DINT	I, Q, M, D, L	<p>Временной интервал (в мс) между вызовом и окончанием выполнения заданного ОВ, и именно для последней законченной обработки определенного ОВ. Если в текущий момент обрабатывается ОВ, для которого запрошено время выполнения, то после первого вызова SFC78 во время выполнения нужного ОВ параметр LAST_ET выводит LAST_ET временной интервал между последним завершенным вызовом ОВ и окончанием обработки заданного ОВ.</p> <p>С каждым последующим вызовом SFC78 во время выполнения нужного ОВ, параметр LAST_ET</p> <ul style="list-style-type: none"> <li>показывает DW#16#FFFF FFFF, если требуемый ОВ уже вызвал SFC78 (если ОВ_NR=0).</li> <li>показывает временной интервал между последним завершенным вызовом ОВ и окончанием обработки заданного ОВ, если этот ОВ не вызывал SFC78 с ОВ_NR=0.</li> </ul> <p>Примечание: Время обработки прерываний, вызванных блоками ОВ с более высоким приоритетом, включаются в значения LAST_ET.</p>

Параметр	Описание	Тип данных	Область памяти	Характеристика
CUR_T	OUTPUT	DINT	I, Q, M, D, L	<p>Время запроса ОВ (относительное время в мс) выполняемого в текущее время ОВ.</p> <p>CUR_T = 0, если заданный ОВ не обрабатывается.</p> <p>Примечание: Системное время - значение счетчика, который считает от 0 до 2.147.483.647 (мс). Счетчик сбрасывается после переполнения.</p>
CUR_RT	OUTPUT	DINT	I, Q, M, D, L	<p>Законченное время выполнения (мс) заданного ОВ. CUR_RT = 0, если ОВ еще не обработан. После обработки данные параметра пересылаются в LAST_RT, а CUR_RT сбрасывается в 0. Примечание: Время обработки прерываний, вызванных блоками ОВ с более высоким приоритетом, не включаются в значения CUR_RT.</p>
CUR_ET	OUTPUT	DINT	I, Q, M, D, L	<p>Время, прошедшее с момента вызова заданного и в настоящий момент обрабатываемого ОВ (в мс) (runtime). CUR_ET = 0, если ОВ еще не обработан. После обработки данные параметра пересылаются в LAST_ET, а CUR_ET сбрасывается в 0. Примечание: Время обработки прерываний, вызванных блоками ОВ с более высоким приоритетом, включаются в значения CUR_ET.</p>
NEXT_ET	OUTPUT	DINT	I, Q, M, D, L	<p>Если последующие вызовы данного ОВ создают очередь до завершения текущего запроса, то параметр NEXT_ET показывает время до запуска - время между текущим временем и временем следующего запроса (в мс) NEXT_ET = 0, если не будет других событий запуска, кроме очередных или обрабатываемых для данного ОВ.</p> <p>WinLC RTX не использует этот параметр.</p> <p>Примечание: Время обработки прерываний, вызванных блоками ОВ с более высоким приоритетом, включаются в значения NEXT_ET.</p>

Значение времени включает в себя также время выполнения для любых вложенных обработок прерываний синхронных ошибок (OB 121, OB 122).

---

**Примечание**

Когда Вы задаете номер OB в параметре OB\_NR, который находится в динамических данных проекта в Вашем CPU, без вызова OS соответствующего OB, или Вы не загрузили его в CPU, то параметр RET\_VAL содержит определенный номер OB, параметр PRIO содержит заданное (если требуется, по умолчанию) значение приоритета для определенного OB и LAST\_RT содержит значение DW#16#FFFF FFFF.

---

**Информация об ошибках**

Класс события Код ошибки	Пояснение
1 ... 102	Номер OB, которому пересылается информация.
W#16#8080	Параметр OB_NR содержит некорректное значение.
W#16#8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 13.6 Запрос данных фактического состояния соединений с помощью SFC87 "C\_DIAG"

### Описание

С помощью SFC87 "C\_DIAG" Вы можете определить фактическое состояние всех S7-соединений и всех текущих возможных S7-соединений (или их отдельных связей).

Соответствующая проверка данных этих соединений позволяет Вам, распознать отказы S7-соединений, наличие фактических S7-соединений и в случае возникновения проблемы получить сообщение для системы управления и в системе визуализации. Контролируемые соединения могут быть как соединениями между отдельными автоматизированными системами так и связью автоматизированной системы с системой управления или с системой визуализации.

### Рабочий режим

SFC87 "C\_DIAG" является асинхронной функцией, что означает, что во время ее выполнения могут произойти несколько вызовов.

Запускайте задание вызовом SFC87 с параметром REQ = 1.

Если задание может немедленно выполняться, SFC возвращает значение 0 в выходном параметре BUZY. Если BUZY = 1, значит задание все еще активно.

### В каких случаях вызывается SFC87?

Для распознавания сбоев S7-соединений вызывайте SFC87 в ОВ циклического прерывания с циклом запуска, например, 10 секунд под управлением операционной системы.

Так как состояние соединения обычно не изменяется, то имеет смысл копировать данные о состоянии соединения в программу пользователя по результатам циклических вызовов только, если состояние соединения изменилось, начиная с последнего вызова (вызывайте функцию с параметром MODE=B#16#02, см. ниже).

### Как вызывается SFC87?

Функция SFC87 "C\_DIAG" имеет 4 рабочих режима, которые показаны в таблице, см. ниже.

MODE (V#16#...)	SFC копирует данные о соединении в программу пользователя	SFC передает подтверждение информации операционной системе
00	Нет	Да
01	Да	Да
02	<ul style="list-style-type: none"> <li>• Да, если произошли изменения в данных о соединении</li> <li>• Нет, если нет изменений в данных о соединении</li> </ul>	Да
03	Да	Нет

Изменение состояния данных о соединении со времени последнего вызова SFC87 (при MODE = V#16#00, 01 или 02) сопровождаются передачей подтверждающей информацией операционной системе.

#### Примечание

Если Вы обрабатываете SFC87 в ОВ циклического прерывания в режиме "копирование по условию" (MODE = V#16#02), Вы должны убедиться, что нет никаких инициализирующих значений в области назначения после холодного старта CPU. Вы можете выполнить это единичным вызовом SFC87 в ОВ 102 в режиме "безусловное копирование с подтверждением" (MODE = V#16#01).

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Управляющий параметр для запроса на активацию с значением REQ = 1: начать выполнение задания, если задание еще не выполняется.
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Назначение задания: <ul style="list-style-type: none"> <li>• V#16#00: SFC не копирует данные о соединении, но передает подтверждающую информацию операционной системе.</li> <li>• V#16#01: вне зависимости от наличия изменения в состоянии соединения SFC копирует всю информацию о соединении в программу пользователя, и передает подтверждающую информацию операционной системе.</li> <li>• V#16#02: только при наличии изменения в состоянии соединения SFC копирует данные о соединении в программу пользователя (если изменений нет, то данные не копируются), и в любом случае передает подтверждающую информацию операционной системе.</li> <li>• V#16#03: вне зависимости от наличия изменения в состоянии соединения SFC копирует всю информацию о соединении в программу пользователя, но не передает подтверждающую информацию операционной системе.</li> </ul>

(продолжение таблицы)

Параметр	Описание	Тип данных	Область памяти	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Возвращаемое значение (код ошибки или состояние задания).
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: Задание пока не завершено.
N_CON	OUTPUT	INT	I, Q, M, D, L	Индекс последней структуры в CON_ARR при условии, что значения .DIS_PCON или .DIS_CON имеют значение TRUE (ИСТИНА). Так в программе пользователя необходимо проверять только первые элементы N_CON CON_ARR. Примечание: Первая структура в поле CON_ARR имеет индекс (index) 1.
CON_ARR	OUTPUT	ANY	I, Q, M, D, L	Область назначения для размещения данных о соединениях. Разрешен только тип данных BYTE. Структура назначается каждому соединению. Выберите размер области назначения, в котором можно разместить все структуры, даже максимальное число возможных соединений для Вашего CPU.

### Организация области назначения CON\_ARR

Область назначения - это поле структуры. Структура назначается каждому соединению. Область не требует начальных значений и может содержать произвольные данные между двумя истинными данными для соединений. Соединения не сортируются по ссылкам.

#### Примечание

Справедливость даты данных о соединении обеспечена, если Вы копируете данные соединения из операционной системы в область назначения.

### Организация структуры

Параметры	Тип данных	Описание
CON_ARR[i]. CON_ID	WORD	Ссылка на соединение, которую Вы должны назначить в NETPRO для данного соединения. W#16#FFFF: неверное назначение, т.е., соединение неконфигурировано. Если CON_ARR[i].DIS_PCON или CON_ARR[i].DIS_CON (см. ниже) установлено, то данное соединение было переконфигурировано или удалено после последнего вызова SFC 87.

Параметры	Тип данных	Описание
CON_ARR[i]. STAT_CON	BYTE	Фактическое состояние S7-соединения Возможные значения: <ul style="list-style-type: none"> <li>• V#16#00: S7-соединение не установлено</li> <li>• V#16#10: Фактическое S7-соединение не разомкнуто</li> <li>• V#16#01: S7-соединение устанавливается</li> <li>• V#16#11: Фактическое S7-соединение в настоящее время разрывается</li> <li>• V#16#02: S7-соединение установлено</li> <li>• V#16#12: Фактическое S7-соединение установлено (одно отдельное соединение установлено)</li> <li>• V#16#13: Фактическое S7-соединение установлено (с двумя отдельными соединениями)</li> </ul>
CON_ARR[i]. PROD_CON	BYTE	Число отдельных соединений в рабочем соединении. Возможные значения: 0, 1, 2, 3.
CON_ARR[i]. STBY_CON	BYTE	Число отдельных соединений в резервном соединении (V#16#FF: нет резервного соединения) Возможные значения: 0, 1, 2, 3. Примечание: Только фактическое соединение может иметь резервирование.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	<ul style="list-style-type: none"> <li>• MODE=V#16#00, 01 или 02: нет изменений в состоянии соединений (элемент структуры STAT_CON) после последнего вызова. Вызов был обработан без ошибки.</li> <li>• MODE=V#16#03: процедура копирования была выполнена без ошибки.</li> </ul>
0001	<ul style="list-style-type: none"> <li>• MODE=V#16#00, 01 или 02: произошли изменения в состоянии (элемент структуры STAT_CON) по крайней мере одного из соединений после последнего вызова. Задание было выполнено без ошибки.</li> <li>• MODE=V#16#03: RET_VAL W#16#0001 не возможно:</li> </ul>
7000	Первый вызов при REQ=0. Задание, определенное в параметре MODE, не может быть выполнено. Параметр BUSY имеет значение 0.
7001	Первый вызов при REQ=1. Задание, определенное в параметре MODE, должно быть инициировано. Параметр BUSY имеет значение 1.
7002	Промежуточный вызов (REQ не имеет значения). Задание все еще выполняется. Параметр BUSY имеет значение 1.
8000	Неверное значение параметра MODE.
8001	Неверное значение параметра CON_ARR.
8002	Размер в параметре CON_ARR слишком мал. SFC не копирует данные в область назначения.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 13.7 Идентификация топологии шины системы ведущего DP-устройства с помощью SFC103 "DP\_TOPOL"

### Описание

Вы можете вызвать функцию SFC103 "DP\_TOPOL" с ее диагностическими репитерами (diagnostics repeaters) для инициации идентификации шинной топологии выбранной системы ведущего DP-устройства (DP master system). Все диагностические репитеры, подключенные к системе ведущего DP-устройства (DP master system), становятся доступными (могут быть адресованы), когда вызывается SFC103.

---

### Примечание

В любое данное время может быть идентифицирована топология только одной системы ведущего DP-устройства (DP master system).

---

Идентификация топологии является необходимым условием для детального отображения местонахождения ошибки в строке сообщения об ошибке. После внесения добавлений или других изменений в физической структуре системы ведущего DP-устройства (DP master system), необходимо вновь идентифицировать топологию с помощью функции SFC103.

К изменениям в физической структуре системы ведущего DP-устройства относятся:

- Изменение длин кабелей
- Добавление или удаление узлов или компонентов с функцией репитера (repeater function)
- Изменение адресов узлов

Каждый диагностический репитер (diagnostics repeater) вводит результирующие данные (result) (PROFIBUS-адрес всех узлов и их абсолютные расстояния до диагностического репитера) в топологическую таблицу (topology table). Вы можете считывать эти значения в STEP 7.

Мы рекомендуем вызывать SFC103 в OB1 или в OB таймерного прерывания (watchdog interrupt). S7-300 не допускает такие вызовы ни в одном OB запуска.

SFC103 записывает сообщения об ошибках, возвращаемые диагностическими репитерами в выходные параметры DPR и DPR1. Если несколько диагностических репитеров в выбранной системе ведущего DP-устройства (DP master system) сообщает об ошибке, то SFC выводит в DPR и DPR1 информацию, относящуюся только к тому диагностическому репитеру, который первым послал сообщение об ошибке. Вызовите SFC13 "DPNRM\_DG" или используйте STEP 7, чтобы считать полную диагностическую информацию. Значения в DPR и DPR1 имеют сигнал NULL, если диагностические репитеры не сообщали об ошибках.



## Принцип работы

Функция SFC103 "DP\_TOPOL" работает асинхронно, что означает, что для ее выполнения требуется несколько вызовов SFC. Вызывайте SFC103 с параметром REQ = 1 для инициализации процесса идентификации топологии DP-шины и с R=1 - для прерывания этого процесса.

Состояние (статус) задания (job status) индицируется в выходных параметрах RET\_VAL и BUSY, см. также значения REQ, RET\_VAL и BUSY для асинхронно работающих SFC.

---

### Примечание

Процесс идентификации топологии шины может потребовать несколько минут.

---

## Идентификация задания

Входной параметр DP\_ID определяет запрос.

При повторном вызове SFC103 "DP\_TOPOL" до завершения предыдущего цикла идентификации шины, последующая реакция SFC зависит от того, является ли новый вызов таким же как предыдущий: Если параметр DP\_ID соответствует заданию, которое все еще не завершено, то вызов SFC интерпретируется как часть последовательности вызовов и значение W#16#7002 выводится в параметр RET\_VAL. Если этот вызов - новое задание, то CPU игнорирует его.

Параметр	Описание	Тип данных	Область памяти	Описание
REQ	INPUT	BOOL	I, Q, M, D, L, constant	REQ=1: Инициация процесса идентификации шинной топологии
R	INPUT	BOOL	I, Q, M, D, L, constant	R=1: Прерывание процесса идентификации шинной топологии
DP_ID	INPUT	INT	I, Q, M, D, L, constant	Идентификатор ID системы ведущего DP-устройства (DP master system), топология шины которого должна быть идентифицирована
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Возвращаемое значение будет содержать код ошибки, если ошибка произошла во время выполнения функции.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Процесс идентификации шинной топологии не закончен.

Параметр	Описание	Тип данных	Область памяти	Описание
DPR	OUTPUT	BYTE	I, Q, M, D, L	PROFIBUS-адрес диагностического репитера (diagnostics repeater), который выдал сообщение об ошибке
DPRI	OUTPUT	BYTE	I, Q, M, D, L	Измеряемая сеть диагностического репитера (diagnostics repeater), который выдал сообщение об ошибке: <ul style="list-style-type: none"> <li>бит 0 = 1: ошибка в сети DP2</li> <li>бит 1 = 1: ошибка в сети DP3</li> <li>бит 4 = 1: временное (temporary) нарушение в сети DP3</li> <li>бит 5 = 1: постоянное (permanent) нарушение в сети DP3</li> </ul>
<p><b>Постоянные ошибки (Permanent errors):</b> Система обнаруживает постоянные ошибки в сети (permanent network errors), которые препятствуют успешному проведению идентификации топологии. Вы можете считать диагностическую информацию с помощью SFC13 "DPNRM_DG" или STEP 7, что позволит Вам получить диагностическую информацию в деталях.</p> <p><b>Временные ошибки (Temporary errors):</b> Система обнаружила временные ошибки в сети (temporary network errors), которые препятствуют успешному проведению идентификации топологии. Причиной ошибки может быть плохой контакт или периодическая ошибка (recurring error). Природа таких нарушений не позволяет выполнить точную локализацию источника ошибки.</p>				

### Информация об ошибках

Когда рассматривается "реальная" ("real") информация об ошибках (код ошибок W#16#8xyz) в следующей ниже таблице, мы рассматриваем два типа событий (event):

- Временные ошибки (Temporary errors): (коды ошибок W#16#80A2 ... 80A4, 80C3, 80C5):  
Ошибки данного типа можно устранить без вмешательства пользователя, с помощью повторного вызова SFC (многократные вызовы, если необходимо).  
Пример такой ошибки: Запрошенный ресурс занят (W#16#80C3).
- Постоянные ошибки (Permanent errors): (коды ошибок W#16#8082, 80B0, 80B2):  
Ошибки данного типа не устраняются автоматически. Новый вызов SFC имеет смысл только после того, как Вы устранили ошибку.  
Пример такой ошибки: Ведущее DP-устройство (DP master) / CPU не поддерживает данную функцию (W#16#80B0).

Код ошибки (W#16#...)	Пояснение
0000	Задание выполнено без ошибок.
7000	Первый вызов с REQ=0. Идентификация шинной топологии не начата. BUSY = 0.
7001	Первый вызов с REQ=1. Идентификация шинной топологии запрошена. BUSY = 1.
7002	Промежуточный вызов (REQ несоответствующий): Идентификация шинной топологии все еще выполняется. BUSY = 1.
7010	Вы запросили прерывание процесса идентификации шинной топологии. Однако, не выполняется работа для устройства с заданным DP_ID. BUSY = 0.
7011	Первый вызов с R=1. Запрос на прерывание идентификации шинной топологии инициирован. BUSY = 1.
7012	Промежуточный вызов: Идентификация шинной топологии все еще не завершена. BUSY = 1.
7013	Конечный вызов: Идентификация шинной топологии была прервана. BUSY = 0.
8082	Система ведущего DP-устройства (DP master system) с заданным значением DP_ID не сконфигурирована.
80A2	Ошибка в процессе идентификации шинной топологии; для получения подробной информации обратитесь к выходным параметрам DPR и DPR1.
80A3	Ошибка в процессе идентификации шинной топологии: "Превышено время таймера" ("Watchdog timeout")
80A4	Коммуникационная ошибка на K-шине
80B0	Ведущее DP-устройство (DP master) / CPU не поддерживает данную функцию.
80B2	Ошибка в процессе идентификации шинной топологии: Диагностический репитер (diagnostics repeater) не обнаружено в выбранной системе ведущего DP-устройства (DP master system).
80C3	Запрошенный ресурс в настоящее время занят. Возможные причины: Вы начали второй цикл идентификации топологии шины (в каждый момент времени допускается только один цикл идентификации топологии шины) или H CPU выполняет действие соединения (connecting) или обновления.
80C5	Система ведущего DP-устройства (DP master system) в настоящее время не доступна.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL



## **14 SFC и SFB для обновления образа процесса и обработки битовых массивов**

## 14.1 Обновление таблицы входов образа процесса с помощью SFC26 "UPDAT\_PI"

### Описание

С помощью SFC 26 "UPDAT\_PI" (update process image [обновить образ процесса]) обновляется таблица входов образа процесса блока OB1 (=раздел 0 образа процесса) или раздел входов образа процесса, определенный с помощью STEP 7.

Если Вы сконфигурировали повторную передачу сигнала об ошибках доступа к входам/выходам для обновления таблицы образа процесса системой, то выбранная таблица образа процесса будет обновляться функцией SFC26 постоянно.

В противном случае SFC26 будет обновлять таблицу образа процесса только тогда, когда выбранный раздел образа процесса не обновляется системой, иными словами:

- когда Вы не поставили этот раздел образа процесса в соответствие OB прерываний или
- когда Вы выбрали раздел 0 образа процесса и заблокировали обновление раздела образа процесса блока OB1 в конфигурации.

### Примечание

Каждый логический адрес, который Вы ставите в соответствие разделу таблицы входов образа процесса с помощью STEP 7, более не принадлежит таблице входов образа процесса блока OB1.

Вызовы SFC26 не оказывают влияния на обновление таблицы входов образа процесса блока OB1 и разделов входов образа процесса, которые Вы назначили OB прерываний.

Параметр	Описание	Тип данных	Область памяти	Характеристика
PART	INPUT	BYTE	I, Q, M, D, L, константа	Номер раздела образа процесса, подлежащего обновлению. Допустимые значения: от 0 до 15 (0 означает образ процесса блока OB1; n, где $1 \leq n \leq 15$ , означает раздел n образа процесса)
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках
FLADDR	OUTPUT	WORD	I, Q, M, D, L	Адрес первого байта, вызвавшего ошибку, если произошла ошибка доступа.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8090	Недопустимое значение входного параметра PART.
8091	Указанный раздел образа процесса не был определен или не находится в разрешенной области таблицы образа процесса в CPU.
8092	Раздел образа процесса обновляется системой с помощью OB, а Вы не сконфигурировали повторную сигнализацию обо всех ошибках доступа к входам/выходам. Образ процесса не был обновлен функцией SFC26 "UPDAT_PI"
80A0	При обновлении была обнаружена ошибка доступа.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 14.2 Обновление таблицы выходов образа процесса с помощью SFC27 "UPDAT\_PO"

### Описание

С помощью SFC27 "UPDAT\_PO" (update process outputs [обновить выходы процесса]) Вы передаете состояния сигналов таблицы выходов образа процесса блока OB1 (= раздел 0 образа процесса) или раздел образа процесса, определенный с помощью STEP 7, в модули вывода.

### Примечание

Каждый логический адрес, который Вы назначаете разделу таблицы выходов образа процесса с помощью STEP 7, больше не принадлежит таблице выходов образа процесса блока OB1.

Вызовы SFC27 не влияют на передачу таблицы выходов образа процесса блока OB1 и разделов выходов образа процесса, которые Вы назначили для OB прерывания.

Параметр	Описание	Тип данных	Область памяти	Характеристика
PART	INPUT	BYTE	I, Q, M, D, L, константа	Номер обновляемого раздела выходов образа процесса. Разрешенные значения: от 0 до 15 (0 обозначает образ процесса OB1, n обозначает раздел n образа процесса, где $1 \leq n \leq 15$ ).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.
FLADDR	OUTPUT	WORD	I, Q, M, D, L	Адрес первого байта, который вызвал ошибку, если произошла ошибка доступа.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8090	Недопустимое значение входного параметра PART.
8091	Заданный раздел образа процесса не был определен или не находится в разрешенной области образа процесса в CPU.
80A0	Во время обновления была обнаружена ошибка доступа.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL



## 14.3 Установка области выходов с помощью SFC79 "SET"

### Описание

Вызов SFC79 "SET" (set range of outputs [установить область выходов]) приводит к следующему результату:

- Устанавливается битовый массив в периферийной области входов/выходов, выбранный с помощью параметров N и SA.
- Соответствующие биты в таблице выходов образа процесса также устанавливаются независимо от того, находятся они в разделе образа процесса или нет.
- Битовый массив должен быть частью периферийной области входов/выходов, назначенной образу процесса.

Если для части выбранного битового массива не подключен модуль, то функция SFC79 все же пытается установить весь битовый массив. Затем она возвращает в RET\_VAL соответствующую информацию об ошибках.

### Примечание

Во время выполнения SFC79 в область входов/выходов всегда записываются целые байты.

Если битовый массив, выбранный с помощью параметров N и SA, не начинается или не заканчивается на границе байта, то вызов SFC79 приводит к следующему результату:

- Биты в первом и последнем байтах, которые должны передаваться в периферийную область входов/выходов и которые не принадлежат выбранному битовому массиву, содержат значения соответствующих битов в таблице выходов образа процесса. Это может привести к непредусмотренным реакциям, таким как запуск электродвигателя или выключение системы охлаждения.
- Биты, принадлежащие выбранному битовому массиву, устанавливаются так, как объяснено выше.

Если Вы присваиваете параметру N значение 0, то вызов SFC79 не оказывает никакого воздействия. Если главное управляющее реле не установлено, то вызов SFC79 не оказывает никакого воздействия.

Параметр	Описание	Тип данных	Область памяти	Характеристика
N	INPUT	INT	I, Q, M, D, L, константа	Количество устанавливаемых битов.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.
SA	OUTPUT	POINTER	P	Указатель на первый устанавливаемый бит.

### Информация об ошибках

Как оценивать информацию об ошибках из параметре RET\_VAL, объясняется в главе 2. Эта глава содержит также общую информацию об ошибках SFC. SFC79 не дает в параметре RET\_VAL какой-либо информации о конкретных ошибках.

## 14.4 Обновление таблицы входов раздела образа процесса в синхронном цикле с помощью SFC 126 "SYNC\_PI"

### Описание

С помощью функции SFC 126 "SYNC\_PI" Вы можете обновлять таблицу входов раздела образа процесса в синхронном цикле. Пользовательская программа, связанная с DP-циклом, может использовать эту функцию SFC для синхронного консистентного обновления данных входов, размещенных в разделе образа процесса (process image partition).

Функция SFC 126 может быть прервана и может вызываться только в OB 61, 62, 63 и 64.

Чтобы обеспечить нормальную обработку SFC 126, необходимо выполнить следующие условия (только для S7-400):

- DP-цикл (DP clock cycle) > приблизительно 5.0 мс
- Время обновления ведомого устройства (slave) < DP-цикла - 4.0 мс

---

### Примечание

Вызов функции SFC 126 "SYNC\_PI" в блоках OB 61...64 допускается, только если Вы назначили обновляемый раздел образа процесса связанному OB в HW Config. Если Вы обновляете таблицу входов раздела образа процесса с помощью SFC 126 "SYNC\_PI", Вы не можете обновлять ее также с помощью SFC 26 "UPDAT\_PI".

---



---

### Внимание (только для S7-400)

Избегайте прямого доступа (например, команды L PIB) к областям раздела образа процесса, которые Вы обрабатываете с помощью функции SFC 126 "SYNC\_PI".

Если Вы игнорируете это правило, то при определенных условиях Вы не сможете получить текущие значения.

---

## Параметры

Параметр	Описание	Тип данных	Диапазон значений	Значение по умолчанию	Характеристика
PART	INPUT	BYTE	1 ... 30		Номер таблицы входов раздела образа процесса, обновляемой в синхронном цикле.
RET_VAL	OUTPUT	INT			Информация об ошибках
FLADDR	OUTPUT	WORD			Адрес первого байта в случае ошибки доступа.

## Информация об ошибках

Класс события Код ошибки	Пояснение
W#16#8090	Недопустимое значение в параметре PART и/или обновление таблицы входов определенного раздела образа процесса не разрешено в этом ОВ. Таблица входов раздела образа процесса не была обновлена.
W#16#8091	Заданный раздел образа процесса все еще не определен или не расположен в доступной области образа процесса в CPU. Таблица входов раздела образа процесса не была обновлена.
W#16#80A0	Во время обновления образа процесса обнаружена ошибка доступа. Входы, которые должны быть обновлены, сброшены в "0".
W#16#80A1	Момент времени обновления расположен после разрешенного для доступа промежутка времени. Таблица входов раздела образа процесса не была обновлена.
W#16#80C1	Момент времени обновления расположен до разрешенного для доступа промежутка времени. Таблица входов раздела образа процесса не была обновлена.
W#16#8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

### Примечание

При использовании функции SFC 126 "SYNC\_PI" для обновления разделов образа процесса для стандартных ведомых DP-устройств, для которых Вы должны определить консистентный диапазон больше, чем 32 байта, также применимы коды ошибок SFC 14 "DPRD\_DAT".

## 14.5 Обновление таблицы выходов раздела образа процесса в синхронном цикле с помощью SFC 127 "SYNC\_PO"

### Описание

С помощью функции SFC 127 "SYNC\_PO" Вы можете обновлять таблицу выходов раздела образа процесса в синхронном цикле. Пользовательская программа, связанная с DP-циклом, может использовать эту функцию SFC для синхронного консистентного обновления данных выходов, размещенных в разделе образа процесса, и с сохранением консистентности пересылать их в I/O устройства.

Функция SFC 127 может быть прервана и может вызываться только в OB 61, 62, 63 и 64.

Чтобы обеспечить нормальную обработку SFC 126, необходимо выполнить следующие условия (только для S7-400):

- DP-цикл (DP clock cycle) > приблизительно 5.0 мс
- Время обновления ведомого устройства (slave) < DP-цикла - 4.0 мс

---

### Примечание

Вызов функции SFC 127 "SYNC\_PO" в блоках OB 61...64 допускается, только если Вы назначили обновляемый раздел образа процесса связанному OB в HW Config. Если Вы обновляете таблицу выходов раздела образа процесса с помощью SFC 127, Вы не можете обновлять ее также с помощью SFC 27 "UPDAT\_PO".

---



### Внимание (только для S7-400)

Избегайте прямого доступа (например, команды L PQB) к областям раздела образа процесса, которые Вы обрабатываете с помощью функции SFC 127. Если Вы игнорируете это правило, то Ваша попытка записи будет неудачной.

---

### Параметры

Параметр	Описание	Тип данных	Диапазон значений	Значение по умолчанию	Характеристика
PART	INPUT	BYTE	1 ...30		Номер таблицы выходов раздела образа процесса, обновляемой в синхронном цикле.

Параметр	Описание	Тип данных	Диапазон значений	Значение по умолчанию	Характеристика
RET_VAL	OUTPUT	INT			Информация об ошибках в возвращаемом значении
FLADDR	OUTPUT	WORD			Адрес первого байта в случае ошибки доступа.

### Информация об ошибках

Класс события Код ошибки	Пояснение
W#16#0001	Предупреждение относительно консистентности. Обновление таблицы выходов определенного раздела образа процесса было распределено на два DP-цикла. Тем не менее, данные в одно ведомое устройство были переданы с соблюдением консистентности.
W#16#8090	Недопустимое значение в параметре PART и/или обновление таблицы выходов определенного раздела образа процесса не разрешено в этом OB. Выходные значения не были переданы в I/O-устройства. Таблица выходов раздела образа процесса не была изменена.
W#16#8091	Заданный раздел образа процесса все еще не определен или не расположен в доступной области образа процесса в CPU. Выходные значения не были переданы в I/O-устройства. Таблица выходов раздела образа процесса не была изменена.
W#16#80A0	Во время обновления образа процесса обнаружена ошибка доступа. Выходные значения не были переданы в I/O-устройства. Таблица выходов раздела образа процесса не была изменена.
W#16#80A1	Момент времени обновления расположен после разрешенного для доступа промежутка времени, или выходные данные не были обновлены ведущим DP-устройством. Выходные значения не были переданы в I/O-устройства. Таблица выходов раздела образа процесса не была изменена.
W#16#80C1	Момент времени обновления расположен до разрешенного для доступа промежутка времени. Выходные значения не были переданы в I/O-устройства. Таблица выходов раздела образа процесса не была изменена.
W#16#8xyy	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

#### Примечание

При использовании функции SFC 127 "SYNC\_PO" для обновления разделов образа процесса для стандартных ведомых DP-устройств, для которых Вы должны определить консистентный диапазон больше, чем 32 байта, также применимы коды ошибок SFC 15 "DPWR\_DAT".

## 14.6 Сброс области выходов с помощью SFC80 "RSET"

### Описание

Вызов SFC80 "RSET" (reset range of outputs [сбросить область выходов]) приводит к следующему результату:

- Сбрасывается битовый массив в периферийной области входов/выходов, выбранной с помощью параметров N и SA.
- Соответствующие биты в таблице выходов образа процесса также сбрасываются независимо от того, находятся они в разделе образа процесса или нет.

Битовый массив должен располагаться в той части периферийной области входов/выходов, которой назначен образ процесса.

Если для части выбранного битового массива не подключен модуль, то SFC80 все же пытается сбросить весь битовый массив. Затем он возвращает в RET\_VAL соответствующую информацию об ошибках.

### Примечание

Во время выполнения SFC80 в периферийную область входов/выходов записываются целые байты.

Если битовый массив, выбранный с помощью параметров N и SA, не начинается или не заканчивается на границе байта, то вызов SFC80 приводит к следующему результату:

- Биты в первом и последнем байтах, которые должны передаваться в периферийную область входов/выходов и которые не принадлежат выбранному битовому массиву, содержат значения соответствующих битов в таблице выходов образа процесса. Это может привести к непредусмотренным реакциям, таким как запуск электродвигателя или выключение системы охлаждения.
- Биты, принадлежащие выбранному битовому массиву, устанавливаются так, как объяснено выше.

Если Вы присваиваете параметру N значение 0, то вызов SFC80 не оказывает никакого воздействия. Если главное управляющее реле не установлено, то вызов SFC80 не оказывает никакого воздействия.

Параметр	Описание	Тип данных	Область памяти	Характеристика
N	INPUT	INT	I, Q, M, D, L, константа	Число сбрасываемых битов.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.
SA	OUTPUT	POINTER	P	Указатель на первый сбрасываемый бит.

### Информация об ошибках

Как оценивать информацию об ошибках из параметра RET\_VAL, объясняется в главе 2. Эта глава содержит также общую информацию об ошибках SFC. SFC80 не дает в параметре RET\_VAL какой-либо информации о конкретных ошибках.

## 14.7 Реализация генератора последовательностей с помощью SFB32 "DRUM"

### Описание

SFB 32 "DRUM" реализует генератор последовательностей с максимальным числом шагов, равным 16. Номер первого шага задается параметром DSP, номер последнего шага – параметром LST\_STEP.

На каждом шаге записываются все 16 выходных битов с OUT0 по OUT15 и выходной параметр OUT\_WORD (в который объединены выходные биты). Выходной бит получает значение либо соответствующего бита заданного вами битового массива OUT\_VAL, либо соответствующего выходного бита из предыдущего шага. Присваиваемое значение зависит от того, как Вы назначите биты маски в параметре S\_MASK (см. следующую таблицу).

SFB32 "DRUM" переходит к следующему шагу, когда на входе JOG появляется положительный фронт по отношению к предыдущему вызову SFB. Если этот SFB уже достиг последнего шага, то при появлении положительного фронта на входе JOG переменные Q и EOD устанавливаются, DCC получает значение 0, и SFB остается на последнем шаге до тех пор, пока на входе RESET не будет установлено значение 1.

Вы можете также назначить параметры так, что переход к следующему шагу будет зависеть от времени. Для этого Вы должны установить параметр DRUM\_EN в 1. Генератор последовательностей переходит к следующему шагу, когда:

- для текущего шага установлен бит события EVENTi и
- истекло запрограммированное для текущего шага время. Это время задается как произведение базы времени DTBP и коэффициента времени, действующего на текущем шаге (из массива S\_PRESET)

---

### Примечание

Остающееся еще на текущем шаге время обработки (DCC) сокращается только тогда, когда установлен соответствующий бит события EVENTi.

---

Если при вызове SFB на входе RESET установлена 1, то генератор последовательностей переходит к шагу, номер которого Вы присвоили входу DSP.

---

### Примечание

Если Вы установили 1 для DRUM\_EN, то Вы можете достичь следующей особой ситуации:

- деблокировки шагов, зависящей только от времени, выбрав  $EVENTi = 1$ , где  $DSP \leq i \leq LST\_STEP$ .
- деблокировки шагов, зависящей только от событий, используя биты событий EVENTi при установке 0 на DTBP.

Вы можете также перейти в генераторе последовательностей к следующему шагу в любое время (даже если DRUM\_EN=1) через вход JOG.

---

Когда блок вызывается в первый раз, Вы должны установить 1 на входе RESET.

Если генератор последовательностей находится на последнем шаге (DSC имеет значение LST\_STEP) и для этого шага истекло время исполнения, то выходы Q и EOD устанавливаются, а SFB остается на последнем шаге до тех пор, пока Вы не установите 1 на входе RESET.

Таймер DRUM работает только в режимах STARTUP и RUN.

Операционная система сбрасывает SFB32 "DRUM" при холодном, но не при теплом рестарте. Если Вы хотите инициализировать SFB32 "DRUM" после теплого рестарта, вызовите его в OB100 с RESET = 1.

Параметр	Описание	Тип данных	Область памяти	Характеристика
RESET	INPUT	BOOL	I, Q, M, D, L, константа	Уровень сигнала 1 сбрасывает генератор последовательностей. При первом вызове блока Вы должны установить RESET в 1.
JOG	INPUT	BOOL	I, Q, M, D, L, константа	Нарастающий фронт (по отношению к последнему вызову SFB) переключает генератор последовательностей на следующий шаг, если только он уже не находится на последнем шаге. Следующий шаг разблокируется в зависимости от значения, которое Вы назначили параметру DRUM_EN.
DRUM_EN	INPUT	BOOL	I, Q, M, D, L, константа	Управляющий параметр, который определяет, возможен ли переход к следующему шагу в зависимости от времени (1: зависящий от времени переход возможен).
LST_STEP	INPUT	BYTE	I, Q, M, D, L, константа	Номер последнего возможного шага. Возможные значения: 1 – 16
EVENT <sub>i</sub> , 1 ≤ i ≤ 16	INPUT	BOOL	I, Q, M, D, L, константа	Бит события номер i (относится к шагу i)
OUT <sub>j</sub> , 0 ≤ j ≤ 15	OUTPUT	BOOL	I, Q, M, D, L	Выходной бит номер j (идентичен биту номер j в OUT_WORD)
Q	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния, который указывает, истекло ли заданное вами время исполнения последнего шага.
OUT_WORD	OUTPUT	WORD	I, Q, M, D, L, P	Выходные биты, объединенные в одну переменную
ERR_CODE	OUTPUT	WORD	I, Q, M, D, L, P	Если во время исполнения SFB появляется ошибка, то ERR_CODE содержит информацию об ошибке.
JOG_HIS	VAR	BOOL	I, Q, M, D, L, константа	(Не имеет значения для пользователя: входной параметр JOG предыдущего вызова SFB)
EOD	VAR	BOOL	I, Q, M, D, L, константа	Идентично выходному параметру Q



Параметр	Описание	Тип данных	Область памяти	Характеристика
DSP	VAR	BYTE	I, Q, M, D, L, P, константа	Номер первого шага. Возможные значения: 1 – 16.
DSC	VAR	BYTE	I, Q, M, D, L, P, константа	Номер текущего шага
DCC	VAR	DWORD	I, Q, M, D, L, P, константа	Остающееся еще на текущем шаге время исполнения в мс (имеет значение только тогда, когда DRUM_EN = 1 и соответствующий бит события = 1)
DTBP	VAR	WORD	I, Q, M, D, L, P, константа	Действительная для всех шагов база времени в мс
PREV_TIME	VAR	DWORD	I, Q, M, D, L, константа	(Не имеет значения для пользователя: системное время предыдущего вызова SFB)
S_PRESET	VAR	ARRAY of WORD	I, Q, M, D, L, константа	Одномерный массив с коэффициентами времени для каждого шага. Рациональный выбор индексов: [1 to 16]. В этом случае S_PRESET [x] содержит коэффициент времени шага x.
OUT_VAL	VAR	ARRAY of BOOL	I, Q, M, D, L, константа	Двумерный массив с выводимыми на каждом шаге значениями, если они не замаскированы с помощью S_MASK. Рациональный выбор индексов: [1 to 16, 0 to 15]. В этом случае OUT_VAL [x, y] содержит значение, которое присваивается выходному биту OUTy на шаге x.
S_MASK	VAR	ARRAY of BOOL	I, Q, M, D, L, константа	Двумерный массив с битами маски для каждого шага. Рациональный выбор индексов: [1 to 16, 0 to 15]. В этом случае S_MASK [x, y] содержит бит маски для y-го выводимого значения на шаге x. Значение битов маски: <ul style="list-style-type: none"> <li>• 0: значение из предыдущего шага присваивается соответствующему выходному биту.</li> <li>• 1: соответствующее значение из OUT_VAL присваивается соответствующему выходному биту.</li> </ul>

**Информация об ошибках**

Если появляется одно из перечисленных в следующей таблице условий, то SFB 32 "DRUM" остается в текущем состоянии, а выход ERR\_CODE устанавливается.

ERR_CODE (W#16#...)	Объяснение
0000	Нет ошибки
8081	Недопустимое значение для LST_STEP
8082	Недопустимое значение для DSC
8083	Недопустимое значение для DSP
8084	Произведение $DCC = DTBP * S\_PRESET[DSC]$ превышает величину $2^{**32-1}$ (примерно 24,86 дня)

## **15 Системные функции для адресации модулей**

## 15.1 Определение логического базового адреса модуля с помощью SFC5 "GADR\_LGC"

### Описание

Канал сигнального модуля определяет соответствующий слот модуля и относительный адрес пользовательских данных модуля. С помощью SFC5 "GADR\_LGC" (convert geographical address to logical address [преобразовать географический адрес в логический]) Вы можете получить соответствующий логический адрес модуля, то есть наименьший адрес входа или выхода.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SUBNETID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор области: <ul style="list-style-type: none"> <li>0, если слот находится в стойке 0 (центральная стойка) или в стойках с 1 по 21 (стойки расширения).</li> <li>идентификатор ведущие DP-устройства соответствующей системы децентрализованной периферии, если слот находится в устройстве децентрализованной периферии.</li> </ul>
RACK	INPUT	WORD	I, Q, M, D, L, константа	<ul style="list-style-type: none"> <li>Номер стойки, если идентификатор области равен 0.</li> <li>номер станции устройства децентрализованной периферии, если идентификатор области &gt; 0.</li> </ul>
SLOT	INPUT	WORD	I, Q, M, D, L, константа	Номер слота.
SUBSLOT	INPUT	BYTE	I, Q, M, D, L, константа	Слот submodule (если submodule не может подключаться, то здесь должен задаваться 0).
SUBADDR	INPUT	WORD	I, Q, M, D, L, константа	Смещение в адресной области пользовательских данных модуля
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.
IOID	OUTPUT	BYTE	I, Q, M, D, L	Идентификатор области: V#16#54: периферийный вход (PI) V#16#55: периферийный выход (PQ) Если модуль смешанный, то SFC дает идентификатор области V#16#54. Если адреса равны, то SFC дает идентификатор V#16#54
LADDR	OUTPUT	WORD	I, Q, M, D, L	Логический базовый адрес модуля.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8094	Никакая подсеть не была конфигурирована с указанным SUBNETID.
8095	Недопустимое значение параметра RACK.
8096	Недопустимое значение параметра SLOT.
8097	Недопустимое значение параметра SUBSLOT.
8098	Недопустимое значение параметра SUBADDR.
8099	Слот не конфигурирован.
809A	Субадрес выбранного слота не конфигурирован.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 15.2 Определение слота модуля, соответствующего логическому адресу, с помощью SFC49 "LGC\_GADR"

### Описание

С помощью SFC 49 "LGC\_GADR" (convert logical address to geographical address [преобразовать логический адрес в географический]) Вы определяете соответствующий логическому адресу слот модуля, а также смещение в пространстве адресов пользовательских данных модуля.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресной области: V#16#54 = периферийный вход (PI) V#16#55 = периферийный выход (PQ) Если модуль смешанный, укажите идентификатор области самого младшего адреса. Если адреса одинаковы, укажите V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес. В случае смешанных модулей укажите младший из двух адресов.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке
AREA	OUTPUT	BYTE	I, Q, M, D, L	Идентификатор области: он указывает, как должны интерпретироваться остающиеся выходные параметры.
RACK	OUTPUT	WORD	I, Q, M, D, L	Номер стойки.
SLOT	OUTPUT	WORD	I, Q, M, D, L	Номер слота
SUBADDR	OUTPUT	WORD	I, Q, M, D, L	Смещение в пространстве адресов пользовательских данных соответствующего модуля.

### Выходной параметр AREA

Выходной параметр AREA указывает, как должны интерпретироваться выходные параметры RACK, SLOT и SUBADDR (см. следующую таблицу).

Значение AREA	Система	Значение RACK, SLOT и SUBADDR
0	S7-400	RACK: номер стойки SLOT: номер слота
1	S7-300	SUBADDR: разность между логическим адресом и логическим базовым адресом.
2	DP	RACK, SLOT и SUBADDR не имеют значения
3	Область S5 P	RACK: номер стойки
4	Область S5 O	SLOT: номер слота
5	Область S5 IM3	SUBADDR: адрес в области S5x
6	Область S5 IM4	

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8090	Заданный логический адрес недопустим или недопустимое значение параметра IOID
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 15.3 Определение всех логических адресов модуля с помощью SFC50 "RD\_LGADR"

### Описание

Вы исходите из одного логического адреса модуля. С помощью SFC 50 "RD\_LGADR" (read module logical addresses [прочитать логические адреса модуля]) Вы получаете все объявленные логические адреса этого модуля. Ранее Вы уже присвоили адреса модулям с помощью STEP 7. SFC50 вводит полученные логические адреса в массив PEADDR или в массив PAADDR в порядке возрастания.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор области: W#16#54: периферийный вход (PI) W#16#55: периферийный выход (PQ)
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Один логический адрес
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке
PEADDR	OUTPUT	ANY	I, Q, M, D, L	Массив для адресов PI, элементы массива должны иметь тип данных WORD.
PECOUNT	OUTPUT	INT	I, Q, M, D, L	Количество возвращенных адресов PI
PAADDR	OUTPUT	ANY	I, Q, M, D, L	Массив для адресов PQ, элементы массива должны иметь тип данных WORD.
PACOUNT	OUTPUT	INT	I, Q, M, D, L	Количество возвращенных адресов PQ

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8090	Заданный логический адрес недействителен или недопустимое значение для параметра IOID.
80A0	Ошибка в выходном параметре PEADDR: Тип данных элементов массива не WORD.
80A1	Ошибка в выходном параметре PAADDR: Тип данных элементов массива не WORD.
80A2	Ошибка в выходном параметре PEADDR: Заданный массив не смог принять все логические адреса.
80A3	Ошибка в выходном параметре PAADDR: Заданный массив не смог принять все логические адреса.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL



## **16 SFC для децентрализованной периферии**

## 16.1 Запуск аппаратного прерывания в ведущем DP-устройстве с помощью SFC7 "DP\_PRAL"

### Описание

С помощью SFC7 "DP\_PRAL" Вы запускаете аппаратное прерывание в ведущем DP-устройстве из программы пользователя в интеллектуальном slave-устройстве. Это прерывание запускает OB40 в ведущем DP-устройстве.

Вы можете идентифицировать причину аппаратного прерывания, используя входной параметр AL\_INFO. Этот идентификатор прерывания передается master-устройству DP, и Вы можете оценивать идентификатор в OB40 (переменная OB40\_POINT\_ADDR).

Запрошенное аппаратное прерывание однозначно определяется входными параметрами IOID и LADDR. Для каждой конфигурированной адресной области в памяти передачи Вы можете в любое время запустить точно одно аппаратное прерывание.

### Как работает SFC

SFC7 "DP\_PRAL" работает асинхронно, другими словами, она выполняется на протяжении нескольких вызовов SFC. Запрос аппаратного прерывания запускают, вызывая SFC7 с REQ=1. Состояние задания отображается выходными параметрами RET\_VAL и BUSY, см. тему "значения параметров REQ, RET\_VAL и BUSY для асинхронных SFC".

Задание завершается тогда, когда завершается выполнение OB40 в ведущем DP-устройстве.

---

### Примечание

Если Вы эксплуатируете ведомое DP-устройство в качестве стандартного slave-устройства, то задание завершается, как только диагностический кадр будет получен ведущим устройством DP.

---

## Идентификация задания

Входные параметры IOID и LADDR однозначно определяют задание.

Если Вы вызвали SFC7 "DP\_PRAL" в slave-устройстве DP и вновь вызываете этот SFC прежде, чем ведущее устройство подтвердило запрошенное аппаратное прерывание, то способ реагирования SFC в значительной степени зависит от того, включает ли новый вызов то же самое задание; если параметры IOID и LADDR соответствуют заданию, которое еще не завершилось, то вызов SFC интерпретируется как продолжающийся вызов, независимо от значения параметра AL\_INFO, и в RET\_VAL вводится значение W#16#7002.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ=1: Запуск аппаратного прерывания в ведущем DP-устройстве, принадлежащем slave-устройству.
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресной области в памяти передачи (с точки зрения ведомые DP-устройства): W#16#54= Периферийный вход (PI) W#16#55= Периферийный выход (PQ) Идентификатором области, принадлежащей смешанному модулю, является младший из двух адресов. Если адреса одинаковые, то задайте W#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Начальный адрес адресной области в памяти передачи (с точки зрения ведомые DP-устройства). Если это область, принадлежащая смешанному модулю, то задайте младший из двух адресов.
AL_INFO	INPUT	DWORD	I, Q, M, D, L, константа	Идентификатор прерывания Он передается OB40, который запустится в ведущем DP-устройстве (переменная OB40_POINT_ADDR). Если Вы используете интеллектуальное slave-устройство с удаленным ведущее устройством, то Вы должны в master-устройстве оценивать диагностический кадр. (См. 170).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время выполнения функции происходит ошибка, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Запущенное аппаратное прерывание еще не было подтверждено ведущее устройством DP.

**Информация об ошибках**

<b>Код ошибки (W#16#...)</b>	<b>Объяснение</b>
0000	Задание было выполнено без ошибок.
7000	Первый вызов с REQ=0. Запрос на аппаратное прерывание не активен; BUSY имеет значение 0.
7001	Первый вызов с REQ=1. Запрос на аппаратное прерывание уже передан master-устройству DP; BUSY имеет значение 1.
7002	Промежуточный вызов (REQ не имеет значения): запущенное аппаратное прерывание еще не было подтверждено ведущее устройством DP; BUSY имеет значение 1.
8090	Неправильный начальный адрес адресной области в памяти передачи.
8091	Прерывание заблокировано (блокировка, конфигурированная пользователем).
8093	Параметры IOID и LADDR адресуют модуль, который не способен к запросу аппаратного прерывания.
80B5	Не разрешен вызов в ведущем DP-устройстве.
80C6	Децентрализованная периферия в настоящее время недоступна.

## 16.2 Синхронизация групп ведомых DP-устройств с помощью SFC11 "DPSYC\_FR"

### Описание

С помощью SFC11 "DPSYC\_FR" Вы можете синхронизировать одну или более групп ведомых DP-устройств.

Функция включает передачу соответствующим группам одной из указанных ниже команд управления или их комбинации:

- SYNC (одновременный вывод и «замораживание» состояний выходов в slave-устройствах DP)
- UNSYNC (отменяет команду управления SYNC)
- FREEZE («замораживание» состояний входов в slave-устройствах DP, считывание «замороженных» входов)
- UNFREEZE (отменяет команду управления FREEZE).

---

### Примечание

Команды управления SYNC и FREEZE также остаются корректными, если Вы выполняете холодный рестарт.

---

### Требования

Прежде чем передавать перечисленные выше команды управления, нужно назначить группам ведомые DP-устройства, используя STEP 7 (см. [/231/](#)). Вы должны знать, какое ведомое DP-устройство какой группе с каким номером назначено, и знать реакции разных групп на SYNC/FREEZE.

### Как работает SFC

SFC11 "DPSYC\_FR" является асинхронной SFC, другими словами, ее выполнение занимает время нескольких вызовов SFC. Задание запускают, вызывая SFC11 с REQ=1.

Состояние задания отображается выходными параметрами RET\_VAL и BUSY, см. тему "значения параметров REQ, RET\_VAL и BUSY для асинхронных SFC".

### Идентификация задания

Если Вы запустили задание SYNC/FREEZE и вновь вызвали SFC11 прежде, чем завершилось первое задание, то реакция SFC зависит от того, предназначен ли новый вызов для того же самого задания. Если входные параметры LADDR, GROUP и MODE согласуются, то вызов SFC интерпретируется как продолжающийся вызов.

## Запись выходов модулей DP

Запись выходов модулей DP запускается следующим образом:

- посредством передачи команд на входы/выходы DP,
- посредством записи таблицы выходов образа процесса в модули (операционной системой в конце OB1 или путем вызова SFC27 "UPDAT\_PO"),
- посредством вызова SFC15 "DPWR\_DAT".

В нормальном режиме ведущее устройство DP циклически (в пределах цикла шины PROFIBUS DP) передает выходные байты на выходы ведомых DP-устройств.

Если Вы хотите, чтобы определенные выходные данные (возможно, распределенные по нескольким slave-устройствам) точно в одно и то же самое время подавались на выходы к процессу, то Вы можете передать соответствующему master-устройству DP команду SYNC, используя SFC11 "DPSYC\_FR".

## Что происходит в результате применения SYNC?

С помощью команды управления SYNC ведомые DP-устройства выбранных групп переключаются в режим синхронизации (Sync), другими словами, ведущее устройство DP передает текущие выходные данные и отдает приказ вовлеченным slave-устройствам DP заморозить свои выходы. С приемом следующих выходных кадров ведомые DP-устройства вводят выходные данные во внутренний буфер, а состояние выходов остается неизменным.

После каждой команды управления SYNC ведомые DP-устройства выбранных групп подают выходные данные из своего внутреннего буфера на выходы к процессу.

Выходы вновь обновляются циклически только тогда, когда Вы передадите команду управления UNSYNC, используя SFC11 "DPSYC\_FR".

---

### Примечание

Если ведомые DP-устройства отдельной группы (групп) в реальном времени не подключены к сети или неисправны во время прохождения команд управления, они не будут переключены в режим SYNC. Соответствующей информации не будет в возвращаемом значении SFC.

---

## Особенности применения ET 200M и ET 200X

Для модулей IM 153-1 (Order No. ...-1AA01 и ...-1AA02 [порядковые номера]), BM 141, BM 142, BM 143 и BM 147 возможно не будет значений (сигналов) на выходах, связанных с I/O-устройствами, после общего сброса, холодного перезапуска и сразу после вызова SFC 11. В таких случаях необходимо вновь вызвать SFC 11 для обеспечения синхронизации.

## Чтение входов модулей DP

Входные данные модулей DP считываются следующим образом:

- посредством применения команд загрузки к входам/выходам DP,
- когда обновляется таблица входов образа процесса (операционной системой в начале OB1 или путем вызова SFC26 "UPDAT\_PI"),
- посредством вызова SFC14 "DPRD\_DAT".

В нормальном режиме ведущее устройство DP циклически (в пределах цикла шины PROFIBUS DP) принимает эти входные данные от своих ведомых DP-устройств и делает их доступными для CPU.

Если Вы хотите, чтобы определенные входные данные (возможно, распределенные по нескольким slave-устройствам) считывались из процесса точно в одно и то же самое время, то передайте соответствующему master-устройству DP команду управления FREEZE, используя SFC11 "DPSYC\_FR".

## Что происходит в результате применения FREEZE?

С помощью команды управления FREEZE вовлеченные ведомые DP-устройства переключаются в режим «замораживания», другими словами, ведущее устройство DP дает команду slave-устройствам DP заморозить текущее состояние входов. Затем оно передает «замороженные» данные во входную область CPU.

После каждой команды управления FREEZE ведомые DP-устройства вновь «замораживают» состояние своих входов.

Ведущее устройство DP вновь циклически принимает текущее состояние входов только после того, как Вы передадите команду управления UNFREEZE с помощью SFC11 "DPSYC\_FR".

---

### Примечание

Если ведомые DP-устройства отдельной группы (групп) в реальном времени не подключены к сети или неисправны во время прохождения команд управления, они не будут переключены в режим FREEZE.

Соответствующей информации не будет в возвращаемом значении SFC.

---

### Консистентность данных

Поскольку SFC11 "DPSYC\_FR" функционирует ациклически и может прерываться классами более высокого приоритета, Вам нужно убедиться, что при использовании SFC11 "DPSYC\_FR" образы процесса совместимы с фактическими входными и выходными данными.

Это гарантируется, если Вы соблюдаете следующие правила консистентности:

- Определите подходящие разделы образа процесса для команд "Синхронизировать (SYNC) выходы" и "Заморозить (FREEZE) входы" (возможно только в S7-400). Вызовите SFC27 "UPDAT\_PO" непосредственно перед первым вызовом задания SYNC. Вызовите SFC26 "UPDAT\_PI" сразу после последнего вызова задания FREEZE.
- В качестве альтернативы: Используйте только прямой доступ к входам/выходам для выходов, включенных в задание SYNC, и для входов, включенных в задание FREEZE. Вы не должны записывать в эти выходы, когда активно задание SYNC, и не должны читать эти входы, когда активно задание FREEZE.

### Использование SFC15 и SFC14

Если Вы используете SFC15 "DPWR\_DAT", то эта SFC должен завершиться прежде, чем Вы передадите задание SYNC соответствующим выходам.

Если Вы используете SFC14 "DPRD\_DAT", то этот SFC должен завершиться прежде, чем Вы передадите задание FREEZE соответствующим входам.

### SFC11 "DPSYC\_FR" и запуск

Только пользователь должен нести ответственность за передачу команд управления SYNC и FREEZE в ОБ запуска.

Если Вы хотите, чтобы выходы одной или более групп были в режиме синхронизации, когда запускается программа пользователя, то Вы должны инициализировать эти выходы во время запуска и полностью выполнить SFC11 "DPSYC\_FR" с командой управления SYNC.

Если Вы хотите, чтобы входы одной или более групп были в режиме FREEZE, когда запускается программа пользователя, то Вы должны выполнить SFC11 "DPSYC\_FR" с командой управления FREEZE полностью для этих входов во время запуска.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Запускаемый уровнем параметр управления REQ=1: запуск задания SYNC/FREEZE
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес ведущие DP-устройства



Параметр	Описание	Тип данных	Область памяти	Характеристика
GROUP	INPUT	BYTE	I, Q, M, D, L, константа	Выбор группы Бит 0 = 1: выбрана группа 1 Бит 1 = 1: выбрана группа 2 : Бит 7 = 1: выбрана группа 8 Вы можете выбрать несколько групп на задание. Значение V#16#0 не разрешено.
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор (ID) задания (кодирование в соответствии с EN 50 170, том 2, PROFIBUS) Бит 0: Резерв (значение 0) Бит 1: Резерв (значение 0) Бит 2 = 1: выполняется UNFREEZE = 0: нет значения Бит 3 = 1: выполняется FREEZE = 0: нет значения Бит 4 = 1: выполняется UNSYNC = 0: нет значения Бит 5 = 1: выполняется SYNC = 0: нет значения Бит 6: Резерв (значение 0) Бит 7: Резерв (значение 0)  Возможные значения: <ul style="list-style-type: none"> <li>• в случае точно одного ID на задание: <ul style="list-style-type: none"> <li>-V#16#04 (UNFREEZE)</li> <li>-V#16#08 (FREEZE)</li> <li>-V#16#10 (UNSYNC)</li> <li>-V#16#20 (SYNC)</li> </ul> </li> <li>• в случае более одного ID на задание: <ul style="list-style-type: none"> <li>-V#16#14 (UNSYNC, UNFREEZE)</li> <li>-V#16#18 (UNSYNC, FREEZE)</li> <li>-V#16#24 (SYNC, UNFREEZE)</li> <li>-V#16#28 (SYNC, FREEZE)</li> </ul> </li> </ul>
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция активна, происходит ошибка, то возвращаемое значение содержит код ошибки. Вы должны оценивать RET_VAL каждый раз после того, как блок выполнен.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Задание SYNC/FREEZE еще не завершилось.

## Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Задание завершилось без ошибок.
7000	Первый вызов с REQ =0. Задание, указанное с помощью LADDR, GROUP и MODE, не активно; BUSY имеет значение 0.
7001	Первый вызов с REQ=1. Задание, указанное с помощью LADDR, GROUP и MODE, было запущено; BUSY имеет значение 1.
7002	Промежуточный вызов (REQ не имеет значения). Активированное задание SYNC/FREEZE все еще активно; BUSY имеет значение 1.
8090	Модуль, выбранный с помощью LADDR, не является ведущим устройством DP.
8093	Этот SFC не разрешен для модуля, выбранного с помощью LADDR (конфигурация или версия ведущие DP-устройства).
8094	Неверный параметр GROUP.
8095	Неверный параметр MODE.
80B0	Группа, выбранная с помощью GROUP не конфигурирована.
80B1	Группа, выбранная с помощью GROUP не назначена этому CPU.
80B2	Задание SYNC, указанное с помощью MODE, не разрешено в группе, выбранной с помощью GROUP.
80B3	Задание FREEZE, указанное с помощью MODE, не разрешено в группе, выбранной с помощью GROUP.
80C2	Временная нехватка ресурсов в ведущем DP-устройстве: Ведущее устройство DP в настоящее время обрабатывает максимальное для CPU количество заданий.
80C3	Это задание SYNC/UNSYNC не может быть активировано в настоящее время, так как в любой момент времени может быть запущено только одно задание SYNC/UNSYNC. Проверьте Вашу пользовательскую программу.
80C4	Это задание FREEZE/UNFREEZE не может быть активировано в настоящее время, так как в любой момент времени может быть запущено только одно задание FREEZE/UNFREEZE. Проверьте Вашу пользовательскую программу.
80C5	Децентрализованная периферия недоступна: Отказ подсистемы DP
80C6	Задание прервано из-за отключения входов/выходов посредством центральным процессором.
80C7	Задание прервано из-за теплого или холодного рестарта в ведущем DP-устройстве.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 16.3 Активация и деактивация ведомых (slave) DP-устройств с помощью SFC12 "D\_ACT\_DP"

### Описание

С помощью SFC12 "D\_ACT\_DP, (deactivate and reactivate configured DP slaves) можно деактивировать и вновь активировать сконфигурированные slave DP-устройства. Кроме того, Вы можете также определять в реальном времени активность или пассивность этих устройств.

Функция SFC12 неприменима для PROFIBUS PA устройств полевого уровня, которые подключаются к ведущей DP-системе посредством DP/PA интерфейса.

### Назначение

Если ведомые DP-устройства сконфигурированы в CPU и при этом отсутствуют в системе в реальном времени или не требуется их использование, тем не менее, CPU будет регулярно продолжать опрашивать эти устройства. SFC12 позволяет деактивировать указанные устройства, после чего CPU перестанет к ним обращаться. Таким образом уменьшается самый короткий цикл DP-шины, уменьшается вероятность ошибок в работе.

### Пример применения

Каждое примененное устройство предполагает только одну из комбинаций выбранных настроек. С помощью этих возможных комбинаций настроек производителем конфигурируется ведомое DP-устройство для создания и поддержания базовой программы пользователя, включающей все возможные опции. С помощью SFC12 можно деактивировать все ведомые DP-устройства, которые не представлены при запуске.

Подобная ситуация существует для набора инструментов, имеющих многочисленные опции, но из которых в реальном времени фактически используются только некоторые из них. Эти инструментальные средства выполнены как ведомые DP-устройства. С помощью SFC12 в программе пользователя можно активировать в реальном времени инструментальные средства, когда они необходимы, и деактивировать, когда они не нужны.

### Как работает SFC

SFC12 "D\_ACT\_DP" является асинхронной SFC, другими словами, ее выполнение занимает время нескольких вызовов SFC. Задание запускают, вызывая SFC12 с REQ=1.

Состояние работы отображается в выходных параметрах RET\_VAL и BUZY; См. раздел "Системное программное обеспечение для S7-300/400. Системные и стандартные функции".

## Идентификация задания

Если Вы начали деактивацию(выключение) или активацию (включение) задания, и Вы вызываете SFC12 вновь, до завершения задания, реакция SFC в значительной степени зависит от того, включает ли новый вызов ту же самую обработку. Если параметр LADDR имеет прежнее значение, то вызов SFC интерпретируется как продолжающийся вызов.

## Деактивация ведомых DP-устройств

Когда Вы деактивируете ведомые DP-устройства с помощью SFC12, выходы процесса устанавливаются в состояние с сконфигурированными значениями или в 0 (безопасное состояние). Назначенное ведущее DP-устройство перестает обращаться к выключенным ведомым DP-устройствам.

Деактивированные ведомые DP-устройства не идентифицируются как дефектные или отсутствующие индикаторами ошибок на ведущем DP-устройстве или CPU.

Изображение состояния процесса по входам деактивированных ведомых DP-устройств изменяется на 0, то есть изменяется точно также, как для сбойных ведомых DP-устройств.

Если Вы используете Вашу программу, чтобы непосредственно обратиться к данным пользователя предварительно деактивированного ведомого DP-устройства, вызывается OB (OB 122) обработки ошибки доступа I/O, и соответствующее стартовое событие выводится в диагностический буфер.

Если Вы пытаетесь обращаться к деактивированному ведомому DP-устройству с помощью SFC (т.е., SFC 59 "RD\_REC"), Вы получите информацию об ошибке в параметре RET\_VAL, касающейся недоступного ведомого DP-устройства.

Деактивация ведомого DP-устройства не вызывает запуск OB (OB 85) обработки ошибки в программе, даже если его входы или выходы принадлежат отображению процесса со стороны системы, которое нужно модифицировать. Никакой вывод информации не будет при этом сделан в диагностический буфер.

Деактивация ведомого DP-устройства не вызывает запуск OB (OB 86) обработки ошибки "отказ стойки", и операционная система также не сделает вывода информации в диагностический буфер.

Если в DP-станции возникнет сбой после того, как Вы деактивировали ее с помощью SFC12, операционная система не обнаружит данный отказ. В результате не будет никакого последующего запуска OB86 или поступления сообщения в диагностический буфер. Отказ станции будет обнаружен только после того, как станция будет вновь активирована, и отказ будет отображен в параметре RET\_VAL.

Если Вы желаете деактивировать ведомые DP-устройства, функционирующие как трансмиттеры (передатчики) в перекрестной связи, мы рекомендуем, чтобы Вы сначала деактивировали приемники (получатели информации), обнаруживающие поступление данных для ведущего DP-устройства.

Деактивируйте DP-устройства-трансмиттеры только после того, как Вы выполните этот шаг.

## Активация ведомых DP-устройств

Когда Вы вновь включаете (активируете) ведомые DP-устройства с помощью SFC12, это конфигурируется назначением соответствующих параметров назначенным ведущим D-устройством (как при восстановлении после сбоя DP-станции). При этом активация DP-устройства завершается, когда ведомое DP-устройство будет способно передать данные пользователя. Активация ведомого DP-устройства не вызывает запуск OB (OB 85) обработки ошибки в программе, даже если его вводы или выходы принадлежат обновляемому отображению процесса со стороны системы. Операционная система также не сделает вывода информации в диагностический буфер. Активация ведомого DP-устройства не вызывает запуск OB (OB 86) обработки ошибки "отказ стойки", и операционная система также не сделает вывода информации в диагностический буфер. Если Вы пытаетесь использовать функцию SFC12, чтобы активировать ведомое DP-устройство, которое было деактивировано и физически отключено (отделено) от DP-шины, светодиод "DP-BUSF" на CPU будет мигать примерно одну минуту. После того, как данный период истечет, SFC должен возвратить сообщение об ошибке W#16#80A2 и выключить индикатор. Ведомое DP-устройство останется выключенным. Если ведомое DP-устройство в дальнейшем будет вновь соединено с DP-шиной, оно должно быть восстановлено (активировано) с помощью функции SFC12.

---

### Примечание

Активация ведомого DP-устройства может быть достаточно трудоемкой операцией. Поэтому, если Вы поелаете отменить запущенный процесс активации, запустите вновь SFC12 с тем же самым значением для LADDR и MODE = 2. Повторяйте вызов SFC12, пока успешная отмена активации не не отобразится в параметре RET\_VAL значением 0.

---

Если Вы желаете активировать ведомые DP-устройства, функционирующие в перекрестной связи, мы рекомендуем, чтобы Вы сначала активировали DP-трансммиттеры (передатчики). Активируйте DP-устройства-приемники только после того, как Вы выполните этот шаг.

## Запуск CPU

В зависимости от режима запуска операционная система CPU ведет себя следующим образом по отношению к ведомым DP-устройствам:

- В холодном и теплом режимах запуска ведомые DP-устройства активируются автоматически.
- В горячем режиме рестарта, состояния DP-устройств остаются неизменными, то есть активированные ведомые DP-устройства остаются активированными и деактивированные ведомые DP-устройства остаются деактивированными.

После запуска CPU, последний циклически обращается ко всем сконфигурированным и не деактивированным ведомым DP-устройствам, в том числе к тем, которые не представлены или не отвечают.

---

### Примечание

ОВ запуска не поддерживают вызовы SFC12.

---

## Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ=1: запуск процесса активации или деактивации
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор задания. Возможные значения ID: <ul style="list-style-type: none"> <li>• 0: Запрос информации о состоянии ведомого DP-устройства (активирован/деактивирован).</li> <li>• 1: Активировать ведомое DP-устройство</li> <li>• 2: Деактивировать ведомое DP-устройство</li> </ul>
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Любой логический адрес ведомого DP-устройства.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит, когда эта функция активна, то возвращаемое значение содержит код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	Код активности: BUSY=1: Задание все еще активно BUSY=0: Задание уже завершено.

## Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Задание завершено без ошибок.
0001	Ведомое DP-устройство активно. (Такой код ошибки возможен только при MODE=0).
0002	Ведомое DP-устройство деактивировано. (Такой код ошибки возможен только при MODE=0).
7000	Первый вызов с REQ =0. Задание, указанное с помощью LADDR не активно; BUSY имеет значение 0.
7001	Первый вызов с REQ=1. Задание, указанное с помощью LADDR, было запущено; BUSY имеет значение 1.
7002	Промежуточный вызов (REQ не имеет значения). Активированное задание все еще активно; BUSY имеет значение 1.
8090	<ul style="list-style-type: none"> <li>• Вы не сконфигурировали модуль с адресом, определенным в LADDR.</li> <li>• Вы задействовали Ваш CPU как I-Slave и задали в LADDR адрес этого "I-Slave".</li> </ul>
8092	Для адресованного ведомого DP-устройства никакое задание активации не выполнено в настоящий момент (этот код ошибки возможен только с MODE = -1.)
8093	Никакому ведомому DP-устройству не назначено адреса, заданного в LADDR, или значение параметра MODE неизвестно.
80A1	Адресованное ведомое DP-устройство не может быть снабжено параметрами (этот код ошибки возможен только с MODE = 1). Примечание: CPU выдает такую информацию, только если в активированном ведомом DP-устройстве возникает отказ во время передачи параметров (параметризации). При некорректной параметризации отдельного модуля SFC возвращает информацию об ошибке в W#16#0000.
80A2	Адресованное ведомое DP-устройство не выдает подтверждение.
80A3	Ведущее DP-устройство не поддерживает данную функцию.
80A4	CPU не поддерживает данную функцию для внешних ведущих DP-устройств.
80A6	Ошибка на слоте ведомого DP-устройства: невозможен доступ к данным пользователя (этот код ошибки возможен только с MODE = 1). Примечание: SFC возвращает данную информацию об ошибке, только в случае отказа активированного ведомого DP-устройства и перед окончанием обработки SFC. SFC возвращает информацию об ошибке W#16#0000, только если отдельный модуль недоступен.
80C1	Функция SFC12 была запущена и обрабатывалась с некорректным логическим адресом (этот код ошибки возможен только с MODE = 1).
80C3	Временная ошибка доступа: CPU в настоящее время обрабатывает максимально возможное количество заданий (этот код ошибки возможен только с MODE = 1 и с MODE = 2).
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 16.4 Чтение диагностических данных ведомые DP-устройства (диагностика slave-устройства) с помощью SFC 13 "DPNRM\_DG"

### Диагностика slave-устройства

Каждое ведомое DP-устройство предоставляет диагностические данные, структурированные в соответствии со стандартом EN 50 170 Volume 2, PROFIBUS. Для считывания этих диагностических данных Вам нужна SFC 13 "DPNRM\_DG".

Принципиальная структура диагностических данных slave-устройства представлена в следующей таблице, а дополнительную информацию Вы найдете в руководствах по slave-устройствам DP.

Байт	Значение
0	Состояние станции 1
1	Состояние станции 2
2	Состояние станции 3
3	Номер master-станции
4	Идентификатор изготовителя (старший байт)
5	Идентификатор изготовителя (младший байт)
6 ...	Дополнительная диагностическая информация, относящаяся к slave-устройству

### Описание

С помощью SFC 13 "DPNRM\_DG" (read diagnosis data of a DP-slave [прочитать диагностические данные ведомые DP-устройства]) Вы читаете текущие диагностические данные ведомые DP-устройства в формате, определенном стандартом EN 50 170 Volume 2, PROFIBUS. Прочитанные данные после безошибочной передачи вводятся в целевую область, указанную в параметре RECORD.

Вы запускаете задание на чтение, присваивая 1 параметру REQ в вызове SFC13.

### Принцип действия

Задание на чтение выполняется асинхронно, т.е. может распространяться на несколько вызовов SFC 13. Состояние задание отображается выходными параметрами RET\_VAL и BUSY, см. тему "значения параметров REQ, RET\_VAL и BUSY для асинхронных SFC".



Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ=1: запрос на чтение
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Сконфигурированный диагностический адрес ведомые DP-устройства.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит, когда эта функция активна, то возвращаемое значение содержит код ошибки. Если ошибок нет, то длина фактически переданных данных вводится в RET_VAL.
RECORD	OUTPUT	ANY	I, Q, M, D, L	Целевая область для прочитанных диагностических данных. Допустимым является только тип данных BYTE. Минимальная длина считываемой записи данных или целевой области равна 6. Максимальная длина отправляемой записи данных равна 240. Стандартные slave-устройства могут предоставить более 240 байтов диагностических данных (максимум до 244 байтов). В этом случае первые 240 байтов передаются в целевую область, и в этих данных устанавливается бит переполнения.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Задание на чтение еще не завершено.

### Входной параметр RECORD

CPU оценивает фактическую длину прочитанных диагностических данных следующим образом:

- Если длина, указанная для RECORD, меньше, чем количество доставленных байтов данных, то данные отбрасываются, и в RET\_VAL записывается соответствующий код ошибки.
- Если длина, указанная для RECORD, больше или равна количеству доставленных байтов данных, то данные принимаются в целевую область, и в RET\_VAL записывается фактическая длина в виде положительного значения.

---

#### Примечание

Вы должны обеспечить, чтобы все фактические параметры RECORD совпадали во всех вызовах, относящихся к заданию.  
Задание однозначно идентифицируется входным параметром LADDR.

---

### Стандартные slave-устройства с более чем 240 байтами диагностических данных

В случае стандартных slave-устройств, в которых количество стандартных диагностических данных находится в диапазоне от 241 до 244 байтов примите во внимание следующие моменты:

- Если длина, указанная для RECORD, меньше 240 байтов, то данные отбрасываются, и соответствующая информация об ошибке вносится в RET\_VAL.
- Если длина, указанная для RECORD, больше или равна 240 байтам, то первые 240 байтов стандартных диагностических данных передаются в целевую область, а в данных устанавливается бит переполнения.

### Выходной параметр RET\_VAL

- Если при исполнении функции происходит ошибка, то возвращаемое значение содержит код ошибки.
- Если при передаче данных не происходит ошибок, то RET\_VAL содержит длину прочитанных данных в байтах в виде положительного числа.

---

#### Примечание

Количество данных для чтения в ведомом DP-устройстве зависит от его диагностического состояния.

---

### Информация об ошибках

Как оценивать информацию об ошибках из параметра RET\_VAL, объясняется в главе 2. Эта глава содержит также общую информацию об ошибках SFC. Специфическая для SFC 13 информация об ошибках является подмножеством информации об ошибках для SFC 59 "RD\_REC",

### Системные ресурсы для S7-400

При вызове SFC 13 "DPNRM\_DG" для задания, не обрабатываемого в данный момент времени, в S7-400 занимают ресурсы CPU (место в памяти). Вы можете вызывать SFC 13 через короткие промежутки времени подряд для нескольких ведомых DP-устройств при условии, что Вы не превышаете максимальное для Вашего CPU количество "одновременно" активных заданий SFC 13. Это максимальное количество Вы найдете в **/101/**.

Если Вы активизируете несколько заданий "одновременно", то все задания будут выполнены, не влияя друг на друга.

Если Вы достигнете ограничения системных ресурсов, то Вы получите сообщение об этом в RET\_VAL. В этом случае повторите задание.

## 16.5 Чтение консистентных данных стандартного ведомые DP-устройства с помощью SFC14 "DPRD\_DAT"

### Консистентность данных

Обратитесь к разделу Обзор системы связи S7 и базовая система связи S7 – консистентность данных.

### Назначение SFC14

Вам понадобится SFC 14 "DPRD\_DAT", так как с помощью команд загрузки, которые выполняют доступ к периферии или к таблице входов образа процесса, Вы можете считывать максимум четыре последовательных байта.

---

#### Примечание

При необходимости Вы можете считывать данные отображения процесса по входу.

Обратитесь к технической документации для определения, поддерживает ли CPU данную функцию.

---

### Описание

С помощью SFC14 "DPRD\_DAT" (read consistent data of a DP standard slave) Вы считываете консистентные данные стандартного ведомые DP-устройства. Длина должна составлять три байта или более четырех байтов, причем максимальная длина фиксирована для каждого конкретного CPU.

Максимальную длину Вы найдете в технических данных своего CPU. Если при передаче данных не было ошибки, то прочитанные данные записываются в целевую область, определяемую параметром RECORD.

Эта целевая область должна иметь ту же самую длину, которую Вы с помощью STEP 7 сконфигурировали для выбранного модуля.

Если Вы читаете из стандартного ведомые DP-устройства, имеющего модульную конструкцию, или с несколькими идентификаторами DP, то Вы можете обратиться к данным только одного модуля или идентификатора DP на каждый вызов SFC14, указывая сконфигурированный начальный адрес.

Параметр	Описание	Тип данных	Область памяти	Характеристика
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Сконфигурированный начальный адрес из области входов (I) модуля, из которой данные будут считываться.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит, когда эта функция активна, то возвращаемое значение содержит код ошибки.
RECORD	OUTPUT	ANY	I, Q, M, D, L	Целевая область для прочитанных данных пользователя. Она должна быть точно такой же длины, какую Вы сконфигурировали с помощью STEP 7 для выбранного модуля. Допустимым является только тип данных BYTE.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибки не было.
8090	Вы не сконфигурировали модуль для заданного логического базового адреса или не учли ограничение на длину консистентных данных.
8092	В ссылке ANY указан тип, отличный от BYTE.
9093	Для заданного в LADDR логического адреса не существует модуля DP, из которого Вы можете считывать консистентные данные.
80A0	Выбранный модуль неисправен.
80B0	Отказ slave-устройства на внешнем интерфейсном модуле DP.
80B1	Длина заданной целевой области не равна сконфигурированной с помощью STEP 7 длине данных пользователя.
80B2	Системная ошибка в случае внешнего интерфейсного модуля DP.
80B3	Системная ошибка в случае внешнего интерфейсного модуля DP.
80C0	Системная ошибка в случае внешнего интерфейсного модуля DP.
80C2	Системная ошибка в случае внешнего интерфейсного модуля DP.
80Fх	Системная ошибка в случае внешнего интерфейсного модуля DP.
87ху	Системная ошибка в случае внешнего интерфейсного модуля DP.
808х	Системная ошибка в случае внешнего интерфейсного модуля DP.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## 16.6 Запись консистентных данных в стандартное ведомое DP-устройство с помощью SFC15 "DPWR\_DAT"

### Консистентность данных

Обратитесь к разделу Обзор системы связи S7 и базовая система связи S7 – консистентность данных.

### Назначение SFC15

Вам понадобится SFC15 "DPWR\_DAT", так как с помощью команд передачи, которые выполняют доступ к периферии или к таблице выходов образа процесса, Вы можете записать максимум четыре последовательных байта

### Описание

С помощью SFC15 "DPWR\_DAT" Вы консистентно передаете указанные в RECORD данные адресованному стандартному slave-устройству DP. Длина передаваемых данных должна составлять три байта или более четырех байтов, причем максимальная длина фиксирована для каждого конкретного CPU. Вы найдете эту информацию в технических данных своего CPU. Данные передаются синхронно, т.е. по окончании SFC задание на запись тоже завершается.

Исходная область должна иметь ту же самую длину, которую Вы сконфигурировали с помощью STEP 7 для выбранного модуля.

Если стандартное ведомое DP-устройство имеет модульную конструкцию, Вы можете получить доступ только к одному модулю ведомые DP-устройства.

Параметр	Описание	Тип данных	Область памяти	Характеристика
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Сконфигурированный начальный адрес из области выходов модуля, в который данные будут записываться.
RECORD	INPUT	ANY	I, Q, M, D, L	Исходная область для записываемых данных пользователя. Она должна быть точно такой же длины, какую Вы сконфигурировали помощью STEP 7 для выбранного модуля. Допустимым является только тип данных BYTE.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если ошибка происходит, когда эта функция активна, то возвращаемое значение содержит код ошибки.

## Информация об ошибках

### Примечание

Если Вы обращаетесь к ведомым DPV1-устройствам, информация об ошибке от этих устройств может быть передана от ведущего DP-устройства к SFC. Для получения более подробной информации по ошибкам, см. тему Получение прерывания от ведомых DP-устройств с помощью SFB 54 "RALRM" STATUS[3].

Код ошибки (W#16#...)	Объяснение
0000	Ошибки не было.
808x	Системная ошибка в случае внешнего интерфейсного модуля DP.
8090	Для заданного логического базового адреса Вы не сконфигурировали модуль или не учли ограничение на длину консистентных данных.
8092	В ссылке ANY указан тип, отличный от BYTE.
9093	Для заданного в LADDR логического адреса не существует модуля DP, в который Вы можете записывать консистентные данные.
80A1	Выбранный модуль неисправен.
80B0	Отказ slave-устройства на внешнем интерфейсном модуле DP.
80B1	Длина заданной исходной области не равна сконфигурированной с помощью STEP 7 длине данных пользователя.
80B2	Системная ошибка на внешнем интерфейсном модуле DP.
80B3	Системная ошибка на внешнем интерфейсном модуле DP.
80C1	Данные предыдущего задания на запись в модуле еще не обработаны модулем.
80C2	Системная ошибка на внешнем интерфейсном модуле DP.
80Fx	Системная ошибка на внешнем интерфейсном модуле DP.
85xu	Системная ошибка на внешнем интерфейсном модуле DP.
8хуу	Общая информация об ошибках, см. Проверка ошибок с выходным параметром RET_VAL

## **17 SFC для связи с помощью глобальных данных**

## 17.1 Передача GD-пакета с помощью SFC60 "GD\_SND"

### Описание

С помощью SFC 60 "GD\_SND" (global data send [передача глобальных данных]) данные пакета глобальных данных (GD-пакета) собираются и затем передаются по маршруту, указанному в GD-пакете. Перед этим GD-пакет должен быть сконфигурирован с помощью STEP 7.

SFC60 "GD\_SND" может быть вызван любом месте программы пользователя.

Вызовы SFC60 не влияют на скорость сканирования, а также сбор и передачу данных системой в точке контроля цикла.

### Возможность прерывания

SFC60 "GD\_SND" может прерываться более высокими классами приоритета. Возможно также, что SFC60 вновь вызывается для того же GD-пакета в более высоком классе приоритета.

Тогда данные собираются и передаются в более высоком классе приоритета. Когда программа возвращается в прерванную SFC, последняя немедленно завершается, а уже собранные данные отбрасываются.

Эта процедура означает, что при обработке самого высокого класса приоритета передаются консистентные данные (в смысле консистентности, определенной для глобальных данных).

### Консистентность данных в GD

Для обеспечения консистентности данных, собранных из различных областей памяти, и переданных данных применяются следующие правила.

Консистентными данными могут быть:

- данные простых типов (бит, байт, слово и двойное слово)
- массив данных типа байт, слово и двойное слово длиной, не превышающей максимума, зависящего от конкретного CPU.

### Обеспечение консистентности для всего GD-пакета

GD-пакет в CPU, передающем данные, имеет структуру, которая не гарантирует автоматически, что собранные данные будут консистентными. Это, например, имеет место, когда пакет состоит из массива байтов, и количество байтов превышает максимальную длину для конкретного CPU.

Если, однако, Вам нужна консистентность для всего GD-пакета, то следуйте в своей программе процедуре, описанной ниже:

- Заблокируйте или задержите появление более приоритетных прерываний и асинхронных ошибок, вызвав SFC 39 "DIS\_IRT" или SFC 41 "DIS\_AIRT".
- Вызовите SFC60 "GD\_SND".
- Вновь разблокируйте более приоритетные прерывания и асинхронные ошибки, вызвав SFC 40 "EN\_IRT" или SFC 42 "EN\_AIRT".



Параметр	Описание	Тип данных	Область памяти	Характеристика
CIRCLE_ID	INPUT	BYTE	I, Q, M, D, L, константа	Номер GD-контура, в котором находится передаваемый GD-пакет. Этот номер задается средствами STEP 7 при конфигурировании глобальных данных. Допустимые значения: от 1 до 16. Максимально возможное количество GD-контуров можно найти в технических данных Вашего CPU.
BLOCK_ID	INPUT	BYTE	I, Q, M, D, L, константа	Номер передаваемого GD-пакета в выбранном GD-контуре. Этот номер задается средствами STEP 7 при конфигурировании глобальных данных. Допустимые значения: от 1 до 3. Максимально возможное количество GD-контуров можно найти в технических данных Вашего CPU.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8081	GD-пакет, выбранный с помощью параметров CIRCLE_ID и BLOCK_ID, не сконфигурирован.
8082	Недопустимое значение для параметра CIRCLE_ID или BLOCK_ID или для обоих параметров.
8083	При выполнении SFC произошла ошибка. Вид ошибки записан в переменную, сконфигурированную для статусной информации. Он может использоваться Вашей программой.
8084	Выполнение SFC было закончено досрочно, так как SFC 60 была вызвана еще раз для того же самого GD-пакета в более высоком классе приоритета (см. "Возможность прерывания").
8085	При записи статусной информации в сконфигурированную для этого переменную произошла ошибка.

#### Примечание

Вам следует после каждого вызова SFC 60 "GD\_SND" проверять состояние соответствующего GD-пакета и, при необходимости, сбрасывать его.

## 17.2 Извлечение принятого GD-пакета с помощью SFC61 "GD\_RCV"

### Описание

С помощью SFC61 "GD\_RCV" (global data receive [прием глобальных данных]) из поступающего кадра GD извлекаются данные точно для одного пакета GD и вводятся в принимаемый пакет GD. Он уже должен быть сконфигурирован с помощью STEP 7.

SFC61 "GD\_SND" может вызываться в любой точке программы пользователя. Вызовы SFC61 не влияют на скорость сканирования и выборку данных системой в контрольной точке цикла.

### Возможность прерывания

SFC61 может прерываться классами более высокого приоритета, но только так, чтобы оставалась гарантированной консистентность данных, определенная для глобальных данных. Если обработка функции прерывается, то возможно, что SFC61 вызывается вновь для того же самого GD-пакета в классе более высокого приоритета.

Тогда данные вводятся в принимаемый GD-пакет в классе более высокого приоритета. Когда программа возвращается к прерванной SFC, она сразу завершается.

### Консистентность данных в GD

Для обеспечения консистентности данных, вводимых в различные области памяти, применяются следующие правила.

Консистентными данными могут быть:

- простые типы данных (бит, байт, слово и двойное слово)
- массивы данных типа байт, слово и двойное слово вплоть до максимальной длины, определенной для принимающего CPU.

### Обеспечение консистентности всего пакета GD

GD-пакет в принимающем CPU имеет структуру, которая не гарантирует автоматически, что его данные происходят из одного и того же кадра. Например, это имеет место, когда пакет состоит из трех элементов GD.

Если, однако, Вы требуете консистентности для всего GD-пакета, то следуйте в Вашей программе процедуре, описанной ниже:

- Запретите или задержите возникновение прерываний более высокого приоритета и асинхронных ошибок, вызывая SFC39 "DIS\_IRT" или SFC41 "DIS\_AIRT".
- Вызовите SFC60 "GD\_SND".
- Разрешите прерывания более высокого приоритета и асинхронные ошибки, вызывая SFC40 "EN\_IRT" или SFC42 "EN\_AIRT".

Параметр	Описание	Тип данных	Область памяти	Характеристика
CIRCLE_ID	INPUT	BYTE	I, Q, M, D, L, константа	Номер GD-контура, в который будет вводиться поступающий GD-пакет. Этот номер задается во время конфигурирования глобальных данных с помощью STEP 7. Разрешенные значения: от 1 до 16. Максимальное число возможных GD-контуров можно найти в технических данных Вашего CPU.
BLOCK_ID	INPUT	BYTE	I, Q, M, D, L, константа	Номер GD-пакета в выбранном GD-контуре, в который будут вводиться поступающие данные. Этот номер задается во время конфигурирования глобальных данных с помощью STEP 7. Разрешенные значения: от 1 до 3. Максимальное число возможных GD-контуров можно найти в технических данных Вашего CPU.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8081	GD-пакет, выбранный с помощью параметров CIRCLE_ID и BLOCK_ID, не сконфигурирован.
8082	Недопустимое значение параметров CIRCLE_ID или BLOCK_ID или обоих параметров.
8083	Во время выполнения SFC произошла ошибка. Тип ошибки вводится в переменную, сконфигурированную для информации о состоянии. Он может оцениваться Вашей программой.
8084	Выполнение SFC было преждевременно завершено, потому что SFC61 был вызван вновь для того же самого GD-пакета в классе более высокого приоритета (см. "Возможность прерывания").
8085	Ошибка произошла при вводе информации о состоянии в сконфигурированную переменную.

### Примечание

После каждого вызова SFC61 Вы должны проверить состояние соответствующего GD-пакета и, в случае необходимости, выполнить его сброс.



## **18 Общий обзор системы связи S7 и базовой системы связи S7**

## 18.1 Различие между блоками системы связи S7 (S7 Communication) и базовой системы связи S7 (S7 Basic Communication)

### Критерии для выбора коммуникативной системы

Кроме связи посредством глобальных данных имеются два других метода обмена данными между программируемыми контроллерами CPU/FM SIMATIC S7:

- Обмен данными с использованием коммуникационных SFB для неконфигурированных S7-соединений.
- Обмен данными с использованием коммуникационных SFB для сконфигурированных S7-соединений.

Метод связи для обмена данными, который Вы выберете, зависит от используемого программируемого контроллера SIMATIC S7 (S7-300, S7-400) и от других параметров обмена данными. Следующая таблица содержит критерии, на которых Вы можете основывать Ваш выбор.

Критерий	Коммуникационные SFB для неконфигурированных S7-соединений	Коммуникационные SFB для сконфигурированных S7-соединений
Доступность блоков	S7-300 и S7-400 как SFC	S7-300 как FB и FC S7-400 как SFB и SFC
Коммуникационные соединения	Соединение не сконфигурировано. Соединение или остается установленным после передачи данных, или разрывается посредством параметра управления. Если временно соединение не может быть установлено, соответствующая работа по передаче данных не будет выполняться.	Соединения являются постоянно сконфигурированными в конфигурации системы.
Переход в STOP-режим	Если CPU, начавший передачу данных, переходит в STOP-режим, то все установленные с ним соединения разрываются.	Соединение поддерживается в STOP-режиме.
Несколько соединений с партнером	В каждый момент времени может быть установлено максимум одно соединение с партнером по связи.	Вы можете установить несколько соединений с партнером.
Диапазон адресов	Модули могут быть адресованы в локальной станции или в MPI-сети.	Модули могут адресоваться в MPI-сети, в PROFIBUS или в Industrial Ethernet.

Критерий	Коммуникационные SFB для неконфигурированных S7-соединений	Коммуникационные SFB для сконфигурированных S7-соединений
Число партнеров по связи	Число партнеров по связи, к которым производится последовательное обращение, не ограничивается доступными коммуникационными ресурсами (см. /70/, /101/). (Связь может утанавливаться и разрываться в процессе выполнения программы).	Число партнеров по связи с одинаковой организацией ограничено доступными коммуникационными ресурсами. Число партнеров по связи также зависит от используемого CPU (см. /70/, /101/).
Максимальная длина данных пользователя для передачи	Гарантируется максимальный объем данных пользователя не менее 76 байтов.	Максимальная длина данных пользователя для передачи зависит от типа блока (USEND / URCV, GET и т.д.) и от партнера по связи (S7-300, S7-400 или M7).
Число передаваемых переменных на каждый вызов блока	Возможна передача только одной переменной.	<ul style="list-style-type: none"> <li>• S7-300: одна переменная</li> <li>• S7-400: максимум 4 переменных</li> </ul>
Классификация блоков	SFC для S7 Basic Communication (базовая система связи S7) являются системными функциями, т.е. они не требуют пользовательской памяти.	SFC для S7 Basic Communication (базовая система связи S7) являются системными функциональными блоками, т.е. они требуют обеспечения экземпляра DB для фактических параметров и статических данных.
Динамическая модификация параметров адресации	Динамическая модификация параметров адресации возможна: по завершении активного задания Вы можете адресовать других партнеров по связи.	S7-300: Вы можете переконфигурировать параметры адресации во время обработки блока. Новые параметры станут доступными, когда завершится предыдущее задание. S7-400: динамическая модификация параметров адресации невозможна: параметры связи остаются зафиксированными, такими, какими были определены при первом вызове блока; они остаются неизменными до следующего теплого или холодного рестарта.

### **Реализация стандарта IEC 61131-5 для S7-400**

Стандарт IEC 61131-5 реализуется с помощью следующих блоков:

- USEND(SFB 8)/URCV(SFB 9)
- BSEND(SFB 12)/BRCV(SFB 13)
- PUT(SFB 15)/GET(SFB 14) соответственно для READ/WRITE
- STATUS(SFB 22)/USTATUS(SFB 23)
- ALARM(SFB 33)
- NOTIFY(SFB 36)
- START (SFB 19), STOP (SFB 20) и RESUME (SFB 21) реализует интерфейс вызова для управляющих функций программы.

### **Реализация стандарта IEC 61131-5 для S7-300**

Стандарт IEC 61131-5 реализуется с помощью следующих блоков:

- USEND(SFB 8)/URCV(SFB 9)
- BSEND(SFB 12)/BRCV(SFB 13)
- PUT(SFB 15)/GET(SFB 14) соответственно для READ/WRITE



## 18.2 Консистентность данных

### Определение

Некоторая область данных, которая может модифицироваться одновременно выполняющимися процессами, называется консистентной областью данных. Таким образом, области данных, большие чем консистентная область данных, могут нести искаженную информацию в целом. То есть, такие области данных могут состоять как из новых так и из старых блоков консистентных данных одновременно.

### Пример

Нарушение консистентности может возникать, если обработка блока связи прервана, например, вызовом ОВ аппаратного прерывания с более высоким приоритетом. Если программа пользователя в этом ОВ изменяет данные, которые уже были частично обработаны в блоке связи, то в итоге перемещенные данные будут состоять:

- частично из данных на момент времени до аппаратного прерывания
- и частично из данных на момент времени после аппаратного прерывания

Это означает, что эти данные неконсистентны (или некогерентны).

### Влияние

Если большие пакеты данных должны передаваться в консистентной форме, то передача не должна прерываться. Выполнение этого требования может увеличивать время реакции на прерывания в CPU. Это означает, что чем больше количество данных, передаваемых в консистентной форме, тем больше будет время реакции на прерывания в системе.

### Консистентность данных в SIMATIC

Если в программе пользователя используется функция связи, например, BSEND/BRCV, которая обращается к общим данным, доступ к этой области данных может координироваться с помощью параметра "DONE". Таким образом может обеспечиваться консистентность данных "коммуникационной" области, которые передаются локально посредством коммуникационного блока (блока связи) в программе пользователя.

Однако, в случае коммуникационных S7-функций связи, например, PUT/GET или write/read (запись/чтение) посредством OP связи размер области консистентных данных должен учитываться уже на этапе программирования или конфигурирования, так как коммуникационные блоки недоступны в

пользовательской программе для устройства назначения (сервер), чтобы можно было обеспечить передачу синхронизированных данных в программе пользователя.

В S7-300 и C7-300 (исключение: CPU 318-2 DP) передаваемые данные копируются последовательно (в консистентной форме) в память пользователя блоками по 32 байтов в контрольных точках (checkpoint) цикла операционной системы. Консистентность данных не гарантируется для больших областей данных. Если требуется консистентность определенных данных, объем данных для передачи в программе пользователя не может превышать 32 байта (максимум 8 байтов, в зависимости от версии).

В S7-400 передаваемые данные обрабатываются не в контрольных точках (checkpoint) цикла операционной системы, а во время фиксированных интервалов времени внутри цикла программы. Консистентность переменных обеспечивается системой.

К таким коммуникационным областям можно обращаться, используя функции PUT/GET или write/read (чтения / записи) переменных.

## Рекомендации

Для получения подробной информации относительно консистентности данных, обратитесь к информации с описанием отдельных блоков, а также к руководству *Communication with SIMATIC* (Средства связи для SIMATIC).

## 18.3 Краткий обзор коммуникационных блоков S7

### Классификация

Для связи в S7 необходимо сконфигурировать соединения. Встроенные функции связи вызываются посредством SFB/FB или SFC/FC в приложении.

Эти блоки можно разбить на следующие категории:

- Коммуникационные SFB/FB для обмена данными
- SFB для изменения режима работы (operating status)
- SFB для запроса режима работы (operating status)
- SFC/FC для запроса режима соединения.

### Доступность коммуникационных блоков для S7-300/400

- Блоки для S7-400 в стандартной библиотеке "Standard Library"
- Загружаемые блоки для S7-300 в библиотеке "SIMATIC\_NET\_CP"

Для запуска блоков S7-300 Вам потребуется SIMATIC NET CP из комплекта S7-300. Для получения подробной информации обратитесь к соответствующей документации.

### Коммуникационные SFB/FB для обмена данными

Коммуникационные SFB/FB для обмена данными используются для передачи данных между двумя партнерами по связи. Если SFB существует только в локальном модуле, то говорят об одностороннем обмене данными. Если SFB/FB присутствует как в локальном, так и в удаленном модулях, то это двусторонний обмен данными.

Блоки S7-400	Блоки S7-300	Описание	Подробное описание
SFB 8 SFB 9	FB 8 FB 9	USEND URCV	Быстродействующий (без квитирования) обмен данными независимо от последовательного выполнения функции связи (URCV) в партнере связи (например, рабочие сообщения и сообщения обслуживания). Это означает, что данные могут быть перезаписаны (заменены) более новыми данными в партнере связи.
SFB 12 SFB 13	FB 12 FB 13	BSEND/ BRCV	Безопасная передача блока данных партнеру связи. Передача данных не завершается, пока функция-премник (BRCV) в партнере связи не приняла данные.

Таблица (продолжение)

Блоки S7-400	Блоки S7-300	Описание	Подробное описание
SFB 14	FB 14	GET	Программно-управляемое чтение переменных без дополнительной функции связи в программе пользователя партнера по связи
SFB 15	FB 15	PUT	Программно-управляемая запись переменных без дополнительной функции связи в программе пользователя партнера по связи
SFB 16	FB 16	PRINT	Передача данных на принтер (только для S7-400)

### SFB для изменения режима работы (только для S7-400)

С помощью SFB для изменения рабочего режима (operating status) Вы можете контролировать рабочий режим удаленного устройства.

Обмен данными с помощью блоков SFB для изменения рабочего режима носит односторонний характер.

Блоки S7-400	Описание	Подробное описание
SFB 19	START	Инициация перезапуска (RESTART) в S7/M7-300/400 или C7-300 CPU, если система находится в режиме STOP.
SFB 20	STOP	Перевод в режим STOP в S7/M7-300/400 или C7-300 CPU, если система находится в режимах RUN, HALT или запуска.
SFB 21	RESUME	Инициация возобновления работы в S7-400-CPU, если он находится в режиме STOP.

### SFB для запроса режима работы (operating status)

С помощью SFB для запроса рабочего режима (operating status) Вы можете получать информацию о рабочем режиме удаленного устройства.

Обмен данными при использовании блока SFB "STATUS" односторонний, тогда как при использовании блока SFB "USTATUS" он носит двусторонний характер.

Блоки S7-400	Описание	Подробное описание
SFB 22	STATUS	Сообщает состояние рабочего режима партнера по связи (M7-300/400 или C7-400-CPU по запросу пользователя).
SFB 23	USTATUS	Принимает информацию о состоянии S7-400-CPU при изменении его рабочего режима, если установлен соответствующий атрибут связи ("послать сообщение о рабочем состоянии").

**SFC/FC для запроса состояния соединения**

Блоки S7-400	Блок S7-300	Подробное описание
SFC 62 CONTROL		Запрос о состоянии соединения, соответствующего экземпляру SFB/FB.
	FC 62 C_CNTRL	Запрос о состоянии соединения для соответствующего ID.

**Совет:**

Вы можете также использовать SFC 87 C\_DIAG для диагностирования состояния соединения (только для S7-400).

**Sample Program (Программа-пример)**

Программа-пример для S7-400, иллюстрирующая использование SFB для S7-соединений поставляется вместе с STEP 7.

Она называется: step7\examples\com\_sfb.

Программа-пример включается в раздел: "..\STEP7\Examples\ZDT01\_10".

**Требования к рабочей памяти для SFB/FB для S7-соединения**

Для нормального функционирования SFB/FB для S7-соединения требуется временная память в рабочей памяти CPU, зависящая от данных пользователя (раздел кода). Размер требуемой памяти показан в следующей таблице:

Блок в S7-300		Требования к рабочей памяти для блоков в байтах
FB 8	USEND	Блок: 4583 байта, Экземпляр: 368 байтов
FB 9	URCV	Блок: 4880 байтов, Экземпляр: 370 байтов
FB 12	BSEND	Блок: 5284 байта, Экземпляр: 372 байта
FB 13	BRCV	Блок: 5258 байтов, Экземпляр: 374 байта
FB 14	GET	Блок: 4888 байтов, Экземпляр: 336 байтов
FB 15	PUT	Блок: 4736 байтов, Экземпляр: 384 байта
FC 62	C_CNTRL	Блок: 546 байтов

**Примечание по поведению при прерываниях**

В S7-300 SIMATIC\_NET коммуникационные блоки могут вызываться только в одном приоритетном классе.

Блок в S7-400		Требования к рабочей памяти для блоков в байтах
SFB 8/ SFB 9	USEND/ URCV	68+длина указанных данных пользователя при первом вызове из SD_1,...,SD_4 / RD_1,...,RD4
SFB 12/ SFB 13	BSEND/ BRCV	54
SFB 14	GET	88+длина указанных данных пользователя при первом вызове из RD_1,...,RD4
SFB 15	PUT	108+длина указанных данных пользователя при первом вызове из SD_1,...,SD_4
SFB 16	PRINT	78+ длина, определенная FORMAT,+ длина указанных данных пользователя при первом вызове из SD_1,...,SD_4
SFB 19	START	52+длина указанного параметра при первом вызове из PI_NAME и ARG.
SFB 20	STOP	48+длина указанного параметра при первом вызове из PI_NAME.
SFB 21	RESUME	52+длина указанного параметра при первом вызове из PI_NAME и ARG.
SFB 22	STATUS	50
SFB 23	USTATUS	50

## 18.4 Краткий обзор блоков для базовой системы связи S7 (S7 Basic Communication)

### Классификация SFC для базовой системы связи S7 (S7 Basic Communication)

Для базовой системы связи в S7 нет необходимости конфигурировать соединения. Встроенные функции связи вызываются посредством SFC в пользовательской программе.

Эти SFC можно разбить на следующие 2 класса:

- SFC для обмена данными между S7 CPU и другими модулями с коммуникационными функциями при условии, что все партнеры по связи принадлежат одной S7-станции (что идентифицируется с помощью префикса "I" ["internal" = внутренний]).
- SFC для обмена данными между S7 CPU и другими модулями с коммуникационными функциями при условии, что партнеры по связи подключены к общей MPI-подсети (что идентифицируется с помощью префикса "X" ["external" = внешний]).

Связь со станциями других подсетей посредством функций SFC для базовой системы связи S7 невозможна.

SFC для базовой системы связи S7 могут запускаться на всех CPU серий S7-300 и S7-400. С этими CPU Вы можете также записывать переменные в CPU серии S7-200 и считывать переменные из них.

### Доступность коммуникационных блоков для S7-300/400

- Блоки для S7-400 в стандартной библиотеке "Standard Library"
- Загружаемые блоки для S7-300 в библиотеке "SIMATIC\_NET\_CP"

Для запуска блоков S7-300 Вам потребуется SIMATIC NET CP из комплекта S7-300. Для получения подробной информации обратитесь к соответствующей документации.

**SFC для внешней (external) связи**

Блок		Подробное описание
SFC 65/ SFC 66	X_SEND/ X_RCV	Безопасная передача блока данных партнеру по связи. Это означает, что передача данных не завершается, пока функция-приемник (X_RCV) в партнере связи не получит все данные.
SFC 67	X_GET	Чтение переменной из партнера по связи без необходимости помещения в него соответствующей функции SFC. Соответствующие функциональные возможности обеспечиваются в партнере по связи операционной системой.
SFC 68	X_PUT	Передача (запись) переменной партнеру по связи без необходимости помещения в него соответствующей функции SFC. Соответствующие функциональные возможности обеспечиваются в партнере по связи операционной системой.
SFC 69	X_ABORT	Прерывание существующего соединения без перемещения данных. Соответствующие ресурсы связи таким образом высвобождаются вновь на обоих концах тракта передачи.

**SFC для внутренней (internal) связи**

Блок		Подробное описание
SFC 72	I_GET	Чтение переменной из партнера по связи без необходимости помещения в него соответствующей функции SFC. Соответствующие функциональные возможности обеспечиваются в партнере по связи операционной системой.
SFC 73	I_PUT	Передача (запись) переменной партнеру по связи без необходимости помещения в него соответствующей функции SFC. Соответствующие функциональные возможности обеспечиваются в партнере по связи операционной системой.
SFC 74	I_ABORT	Прерывание существующего соединения без перемещения данных. Соответствующие ресурсы связи таким образом высвобождаются вновь на обоих концах тракта передачи.

**Sample Programs (Программы-примеры)**

Две программы-примеры для базовой системы связи S7, иллюстрирующие использование SFB для S7-соединений поставляются вместе с ПО STEP 7. Программы-примеры включаются в разделы: "STEP7\Examples\com\_SFC1" и "STEP7\Examples\com\_SFC2".

**Максимальный размер данных пользователя**

Коммуникационные SFC для неконфигурированного S7-соединения интегрированы во все CPU серий S7-300 и S7-400. Все коммуникационные SFC гарантируют передачу 76 байтов данных пользователя (параметр SD и RD).



## Соединение с партнером по связи

Коммуникационные SFC для несконфигурированного S7-соединения обеспечивают связь, пока находятся в состоянии выполнения. В зависимости от значения, которое Вы назначите для входного параметра CONT, связь остается или размыкается после завершения обмена данными. Это означает, что связь имеет следующие характеристики:

- Число партнеров по связи, к которым обеспечен последовательный допуск (обращение в порядке очереди), выше числа партнеров, с которыми возможно одновременное установление связи (Это число зависит от конкретного CPU, см. /70/, /101/).
- Если в данный момент не может быть установлено связи с партнером по причине занятости коммуникационных ресурсов (с локальным CPU или с партнером по связи), это отражается в параметре RET\_VAL. В этом случае Вы должны запустить задание на выполнение вновь позднее в подходящий момент времени. Однако нет гарантии, что и позднее связь с партнером будет установлена успешно. Если это необходимо, проверьте использование коммуникационных ресурсов в Вашей программе и используйте CPU с большими ресурсами.

Существующие соединения с коммуникационными SFB для конфигурируемых S7-соединений не могут использоваться коммуникационными SFC для неконфигурируемых S7-соединений. В запущенном на выполнение задании установленная связь может использоваться только для данного конкретного задания. Другие задания, включая связанные с тем же самым партнером по связи, могут быть выполнены только по завершении текущего задания.

---

### Примечание

Если Ваша программа содержит несколько заданий, в том числе связанных с одним и тем же партнером по связи, разнесите вызовы SFC, для которых W#16#80C0 присутствует в параметре RET\_VAL, в удобные моменты времени.

---

## Идентификация задания

Если Вы инициировали передачу данных или разрываете соединение с помощью одной из коммуникационных функций SFC для несконфигурированных S7-соединений и затем вновь вызываете эту же функцию SFC прежде, чем текущее задание будет завершено, то поведение SFC будет определяться тем, включает ли новый вызов то же самое задание или нет. Следующая таблица показывает, какие входные параметры определяют работу для каждой функции SFC. Если параметры для вызываемой функции одинаковы с параметрами функции, которая еще не завершена, то вызов SFC интерпретируется как продолжающийся вызов.

Блок		Параметры для идентификации заданий
SFC 65	X_SEND	DEST_ID, REQ_ID
SFC 67	X_GET	DEST_ID, VAR_ADDR
SFC 68	X_PUT	DEST_ID, VAR_ADDR
SFC 69	X_ABORT	DEST_ID
SFC 72	I_GET	IOID, LADDR, VAR_ADDR
SFC 73	I_PUT	IOID, LADDR, VAR_ADDR
SFC 74	I_ABORT	IOID, LADDR

### Реакция на прерывания

Работа коммуникационных функций SFC для неконфигурированных S7-соединений может прерываться вызовом OB с более высоким приоритетом. Если такая же функция SFC с идентичным заданием вновь вызывается в таком OB, то этот второй вызов отменяется и дается соответствующий ввод в параметр RET\_VAL. Затем продолжается выполнение прерванной функции SFC.

### Доступ в рабочую память CPU

Независимо от количества передаваемых данных пользователя коммуникационные функции операционной системы обращаются к наибольшим областям рабочей памяти CPU, так что время реагирования на прерывание не увеличивается при использовании функций связи.

В зависимости от того, какой Вы установите максимальный цикл загрузки по связи средствами STEP 7, к рабочей памяти можно обратиться несколько раз во время выполнения задания коммуникационными функциями операционной системы.

### Переход к режиму STOP

Если CPU, который выполняет задание (и, следовательно, поддерживающий соединение) переходит в режим STOP в то время, как идет передача данных, то все соединения, поддерживаемые данным CPU, будут разорваны.

## **Выполнение изменений в программе**

Все части Вашей программы, которые незамедлительно влияют на вызовы коммуникационных функций SFC для неконфигурированных S7-соединений, должны измениться только в режиме STOP. Это касается, в частности, удаления FCS, FBS, или OB, содержащих вызовы коммуникационных функций SFC для неконфигурированных S7-соединений.

После изменения программы Вы должны выполнить теплый или холодный рестарт.

Невыполнение этих правил может привести к оставлению назначенных, но не существующих физических ресурсов, в результате чего программируемый контроллер может оказаться впоследствии в неопределенном состоянии.



## **19 Система связи S7 (S7 Communication)**

## 19.1 Общие параметры блоков SFB/FB и функций SFC/FC системы связи S7 (S7 Communication)

### Классификация

Параметры коммуникационных SFB/FB для сконфигурированных S7-соединений могут быть разбиты на 5 категорий в соответствии с их функциональными особенностями:

1. Параметры управления (управляющие параметры [control parameters]), служащие для активации блока.
2. Параметры адресации (addressing parameters), служащие для адресации удаленных партнеров по связи.
3. Параметры для передачи (send parameters), указывающие на области, данные из которых должны быть переданы удаленному партнеру по связи.
4. Параметры для приема (recieve parameters), указывающие на области данных, в которых должны быть размещены данные, принятые от удаленного партнера по связи.
5. Параметры состояния (status parameters), которые используются для мониторинга результатов обработки блока – завершена ли обработка без ошибок или нет и для анализа возникших ошибок.

### Параметры управления (control parameters)

Обмен данными будет активирован только в том случае, если соответствующие параметры управления будут иметь соответствующие состояния (например, установлены), если вызывается SFB/FB или когда эти состояния параметров (состояния сигналов) особым образом изменяются с момента предыдущего вызова SFB/FB (например, считанное значение параметра имеет характер переднего фронта импульсного сигнала).

---

#### Примечание для S7-300

Для первого вызова установите параметр REQ в состояние FALSE (ЛОЖЬ).

## Параметры адресации

Параметр	Описание
ID	Ссылка на идентификатор локального соединения (определено STEP7-конфигурацией соединения).
R_ID	Используйте параметр R_ID для определения режима, при котором посылаемый и получаемый SFB являются общими: параметр R_ID должен быть идентичен в SFB/FB в точке передачи и в SFB в точке приема. Это обеспечивает связь отдельных пар SFB/FB через одно и то же логическое соединение. <ul style="list-style-type: none"> <li>• R_ID должен быть определен в формате DW#16#wxyzWXYZ.</li> <li>• Пары блоков логического соединения, которые определены в R_ID, должны быть уникальны для этого соединения.</li> </ul>

Параметр PI\_NAME должен быть описан только в соответствующих SFB (только для S7-400).

### Примечание

Для S7-300:

Вы можете изменять значения параметров адресации ID и R\_ID в то время когда они активны. Новые значения параметров вступают в силу с началом каждого нового задания после того, как предыдущая работа была завершена. Здесь Вы можете связывать множество FB пар с одним экземпляром DB.

Совет:

Вы имеете следующие возможности сохранить образцы DB и следовательно рабочую память:

1. С помощью переменных ID Вы можете использовать несколько соединений посредством одного экземпляра DB.
2. С помощью переменных R\_ID Вы можете определять несколько тождественных пар FB для передачи и приема данных в одном задании.
3. Вы можете комбинировать пункты 1 и 2.

Контролируйте корректность новых значений параметров после выполнения последнего задания. Если Вы активируете работу по передаче данных, параметр R\_ID для FB передачи и приема должны быть одинаковы.

Для S7-400:

параметры адресации ID и R\_ID проверяются только при первом вызове блока (фактические параметры или предопределенные значения из образца DB). Следовательно, первый вызов определяет связь (соединение) с удаленным партнером до момента, когда будет выполняться следующий теплый или холодный рестарт.

## Параметры состояния (status parameters)

Параметры состояния Вы можете использовать для проверки результатов обработки блока – находится ли блок в активном состоянии, завершена ли обработка без ошибок или нет и для анализа возникших ошибок.

---

### Примечание

Параметры состояния справедливы только в одном цикле, а именно в интервале между первой инструкцией вызова SFB/FB до следующего вызова SFB/FB.

---

## Параметры для передачи (send) и для приема (recieve parameters)

Если Вы не используете все параметры для передачи или приема блока SFB/FB, то первый неиспользуемый параметр должен быть "нуль"-указателем (NIL pointer) (см. /232/) и параметры должны следовать один за другим без промежутков.

---

### Примечание для S7-400

Во время первого вызова указатель ANY определяет максимальное число данных пользователя, которое может быть передано при выполнении задания. То есть, в рабочей памяти CPU создается буфер передачи/приема для обеспечения консистентности данных. Этот буфер занимает до 480 байт в ОЗУ. Мы рекомендуем выполнять первый вызов в ОВ теплого или холодного перезапуска, если блок не перезагружается при вызове SFB во время RUN-режима CPU.

При последующих вызовах Вы можете посылать / получать любое количество данных, однако, не больше, чем при первом вызове. Параметры BSEND и BRCV для SFB являются исключением из этого правила. С помощью этих функций Вы можете передавать до 64 Кб в одном задании (см. разделы: "Передача сегментированных данных с помощью SFB 12 "BSEND" и "Прием сегментированных данных с помощью SFB 13 "BRCV").

---

С помощью SFB/FB для двустороннего обмена данными:

- Число используемых параметров SD<sub>i</sub> и RD<sub>i</sub> должно быть одинаково для передающей и принимающей сторон.
- Типы данных используемых параметров SD<sub>i</sub> и RD<sub>i</sub>, которые являются общими, должны быть одинаковы для передающей и принимающей сторон.
- Количество данных, пересылаемых в соответствии с параметром SD<sub>i</sub> не должно превышать область памяти, отводимую для приема в соответствии с параметром RD<sub>i</sub> (за исключением параметров BSEND/BRCV).



Значения ERROR = 1 и STATUS = 4 показывают наличие нарушения вышеуказанных правил.

Максимальная длина данных пользователя зависит от того, какого типа удаленный партнер по связи: S7-300 или S7-400.

### Размер данных пользователя

Количество передаваемых данных пользователя с помощью SFB/FB, USEND, URVCV, GET и PUT не должно превышать установленной максимальной величины. Максимальное количество данных пользователя зависит от:

- от используемого типа блока;
- партнера по связи.

Гарантированное количество данных пользователя для SFB/FB при 1-4 переменных в зависимости от типа блока и партнера по связи представлено в следующей таблице:

Блок	Партнер: S7-300/C7-300	Партнер: S7-400/M7 M7 - M7
PUT / GET	160 байтов	400 байтов
PUT / GET	160 байтов	440 байтов
BSEND / BRCV	32768 байтов	65534 байта

Дополнительную информацию по размеру данных пользователя можно найти в технической документации на соответствующие CPU.

### Точный размер данных пользователя

Если данных по размеру передаваемых данных пользователя, указанных выше, не достаточно для установленной точного максимального значения в байтах, выполните следующее:

1. Сначала найдите размер блока данных для условий соединения в следующей таблице:

Ваш CPU	Удаленный CPU	Размер блока данных в байтах:
S7-300	Любой	240 (S7-300)
S7-400	S7-300 / C7-300	240 (S7-400)
S7-400	S7-400 или CPU 318	480
S7-400	Модуль M7	480
Модуль M7	Модуль M7	960

2. Используя полученное значение в следующей таблице, найдите искомое максимальное количество данных пользователя в байтах. Оно задается четным значением размера области памяти для параметров: SD\_i, RD\_i, ADDR\_i.

		Число используемых параметров SD_i, RD_i, ADDR_i			
Размер блока данных	SFB/FB	1	2	3	4
240 (S7-300)	PUT/GET/USEND	160	-	-	-
240 (S7-400)	PUT	222	218	214	210
	GET	212	196	180	164
480	PUT	462	458	454	450
	GET	452	436	420	404
	USEND	452	448	444	440
960	PUT	942	938	934	930
	GET	932	916	900	884
	USEND	932	928	924	920

## 19.2 Подпрограмма запуска SFB для сконфигурированного S7-соединения

### Требования

Требования для S7-400:

- Описания соединения (SDB) присутствуют в модулях.
- Сконфигурированные соединения установлены.
- Фактический параметр для ID соответствует ID сконфигурированного соединения для каждого SFB.

### Теплый и холодный перезапуск

Во время теплового и холодного перезапуска все SFB устанавливаются в состояние NO\_INIT. Действующие (фактические) параметры, сохраняемые в экземпляре DB, не изменяются.

### Теплый и холодный перезапуск для SFB при двустороннем обмене данными

В общем случае для двух модулей с SFB при двустороннем обмене данными не выполняются одновременно теплый или холодный рестарт. Поведение SFB при теплом или холодном рестарте определяется правилами, указанными ниже:

Блоки для приема данных (URCV, BRCV) при теплом или холодном рестарте ведут себя следующим образом:

- Если SFB принял задание, но не подтвердил его выполнение во время теплового или холодного рестарта, он генерирует сообщение о прекращении обработки последовательности данных (CFB, BRCV) и затем немедленно выполняет переход в состояние NO\_INIT.
- Для SFB BRCV возможна ситуация, когда другой сегмент данных будет получен, несмотря на то, что было послано сообщение об аварийном прекращении обработки последовательности данных. Он будет сброшен локально.
- SFB URCV немедленно переходит в состояние NO\_INIT.

Блоки для передачи данных (USEND, BSEND) при теплом или холодном рестарте ведут себя следующим образом:

- Если SFB BSEND начал выполнять задание, и задание еще не завершено, когда инициализирован теплый или холодный рестарт, то генерируется сообщение о прекращении обработки данных. После этого немедленно выполняется переход в состояние NO\_INIT. Подтверждение, которое приходит позднее, отбрасывается локально.
- Если SFB BSEND успел послать или получить сообщение о прекращении работы к моменту запроса теплого или холодного рестарта, то он немедленно переходит в состояние NO\_INIT.
- Во всех других случаях и всякий раз, когда SFB только посылает сообщения (например, SFB USEND), локальная работа прерывается, и SFB немедленно выполняет переход в состояние NO\_INIT.

### **Теплый и холодный перезапуск для SFB при одностороннем обмене данными**

Предполагается, что сервер партнера по связи находится в активном рабочем состоянии после того, как были установлены соединения, другими словами, сервер может в любое время обрабатывать задания или выводить сообщения.

SFB, которые посылают задания и ожидают подтверждения, реагируют на полный рестарт следующим образом:

Текущая обработка задания прерывается и SFB немедленно переходит в состояние NO\_INIT. Если подтверждение для задания, переданного до теплого или холодного рестарта, приходит позже, оно игнорируется локально.

Новое задание может быть передано прежде, чем получено квитирование более ранней обработки.

SFB, которые посылают или получают сообщения, реагируют следующим образом:

- Текущая обработка прерывается, и SFB немедленно переходит в состояние NO\_INIT.
- Для SFB USTATUS сообщения, которые прибывают во время нахождения в состоянии NO\_INIT и заблокированном (DISABLE) состоянии, отбрасываются локально.

### **Реакция на горячий перезапуск**

SFB для S7-соединения переходят в состояние NO\_INIT только во время теплого или холодного рестарта. Это означает, что они реагируют подобно функциональным блокам пользователя, которые могут продолжить выполняться после горячего рестарта.

### **Реакция на сброс памяти**

Сброс (RESET) памяти всегда разрывает все соединения. Так как теплый или холодный рестарт – это единственно возможные способы запуска для программы пользователя после сброса памяти, все блоки SFB для S7-соединений (если они все еще установлены) переходят в состояние NO\_INIT и инициализируются. Блоки партнера по связи в модуле, память которого не была сброшена, реагируют переходом в состояния IDLE (не занят), ENABLE (разрешен) или DISABLE (блокировка) на разрыв соединения.

## 19.3 Реакция блоков SFB на различные ситуации в системе

В следующем разделе описано, как SFB для S7-связи в S7-400 реагируют на различные ситуации в системе.

### Реакция на разрыв соединения

Состояние соединений, размещенных в экземплярах SFB, проверяются.

Если соединение разрывается, реакция SFB зависит от его внутреннего состояния.

Если разрыв соединения происходит в то время, когда блок находится в состояниях IDLE (не занят) или ENABLE (разблокирован), то SFB реагирует следующим образом:

- SFB переходит в состояние ERROR (ошибка) и выводит идентификатор ошибки ID "Communication problems" ("Ошибки связи") в выходные параметры: состояния STATUS и ошибки ERROR.
- При следующем вызове блок возвращается к первоначальному состоянию и вновь проверяет соединение.

Если разрыв соединения происходит в то время, когда коммуникационный SFB не находится в состояниях IDLE (не занят) или ENABLE (разблокирован), то блок реагирует следующим образом:

- Блок прерывает обработку, немедленно или при следующем вызове переходит в состояние ERROR (ошибка) и выводит идентификатор ошибки ID "Communication problems" ("Ошибки связи") в выходные параметры: состояния STATUS и ошибки ERROR.
- При следующем вызове блок переходит в состояния IDLE (не занят), ENABLE (разрешен) или DISABLE (блокировка). В состояниях IDLE (не занят) и ENABLE (разрешен) соединение проверяется вновь.

Эта процедура также будет выполнена, если соединение было вновь восстановлено.

### **Реакция на выключение питания**

Выключение питания при наличии батарейного резервирования питания сопровождается перезапуском, при котором все установленные соединения разрываются. Пункты, рассмотренные выше, следовательно, относятся ко всем подключенным блокам.

Если имеет место выключение питания при наличии батарейного резервирования питания, сопровождающееся автоматическим теплым или холодным рестартом, то рассмотренные пункты, касающиеся завершения соединений и теплого или холодного рестарта справедливы.

В частном случае автоматического теплого или холодного перезапуска без батарейного резервирования питания с автоматическим выполнением сброса памяти после восстановления питания, SFB для S7-связи реагируют согласно описанию в разделе "Подпрограмма запуска SFB для S7-соединений".

### **Реакция на изменение рабочего режима**

При изменениях рабочего режима между состояниями STOP (стоп), START (запуск), RUN (выполнение) и HOLD (задержка) коммуникационный SFB остается в текущем состоянии (исключение: во время теплого или холодного рестарта, его состояние изменяется на NO\_INIT). Это справедливо и в отношении SFB для одностороннего, и в отношении SFB для двустороннего обмена данными.

### **Интерфейс ошибки в программе пользователя**

Если ошибка происходит во время обработки коммуникационного SFB, это всегда влияет на состояние параметра ERROR (ошибка). Выходной параметр ERROR (ошибка) устанавливается в 1, и в то же время соответствующий идентификатор ошибки ID вводится в выходной параметр STATUS (состояние). Вы можете оценивать эту информацию об ошибке в Вашей программе.

Примеры возможных ошибок:

- Ошибка во время сбора (collecting) посылаемых данных.
- Ошибка при копировании принятых данных в область назначения (например, попытка обратиться к DB, который не существует).
- Размер области пересылаемых данных не соответствует длине области, назначения, определенной в партнере SFB.

## 19.4 Некоординированная передача данных посредством SFB 8/FB 8 "USEND"

### Описание

SFB8 "USEND" передает данные блоку SFB/FB типа "URCV" удаленного партнера по связи. Функция выполняется без координации с SFB/FB партнера. Это означает, что передача данных выполняется без подтверждения со стороны SFB/FB партнера по связи.

S7-300: Данные передаются при нарастающем фронте сигнала на входе REQ. Параметры R\_ID, ID и SD\_1 передаются при каждом нарастающем фронте сигнала на входе REQ. После выполнения задания Вы можете назначить новые значения для параметров R\_ID, ID и SD\_1.

S7-400: Данные передаются при нарастающем фронте сигнала на входе REQ. Передаваемые данные указываются параметрами SD\_1 – SD\_4, но не обязательно использовать все четыре параметра передачи.

Однако Вам нужно убедиться в том, что области, определенные параметрами SD\_i и RD\_i,  $1 \leq i \leq 4$  (RD\_i принадлежит соответствующему SFB/FB "URCV" партнера по связи) совпадают друг с другом:

- по числу,
- по размеру,
- по типу данных.

Параметр R\_ID должен быть одинаковым для обоих SFB.

Успешное завершение передачи данных индицируется единичным значением параметра состояния DONE: DONE = 1.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L,	Параметр управления «запрос».
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
R_ID	INPUT	DWORD	I, Q, M, D, L, константа	Параметр адресации R_ID См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: передача не началась или еще продолжается; 1: данные переданы.



Параметр	Описание	Тип данных	Область памяти	Характеристика
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
S7-300: SD_1  S7-400: SD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	M, D, T, Z  I, Q, M, D, T, C	Указатель на i-ю область передаваемых данных. Разрешенные типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER Если параметр ANY указывает на DB, то последний должен быть определен (например, P# DB10.DBX5.0 Byte 10).

### Информация об ошибках

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не может вступить в силу, так как предыдущее задание еще не закончено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Ошибки связи, например: <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное);</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP);</li> <li>связь с партнером пока не установлена.</li> </ul>
1	4	Ошибка в указателях областей передаваемых данных SD_i, включая длину данных или тип данных.
1	10	Доступ к локальной памяти пользователя невозможен (например, обращение к DB, который был удален).

ERROR	STATUS (десятичное число)	Объяснение
1	12	При вызове SFB: <ul style="list-style-type: none"> <li>• был задан экземпляр DB, который не принадлежит SFB "USEND"</li> <li>• был задан не экземпляр DB, а общедоступный DB</li> <li>• экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>
1	18	R_ID уже существует в идентификаторе (ID) соединения.
1	20	Не хватает доступной рабочей памяти. <ul style="list-style-type: none"> <li>• H-System: первый вызов SFB во время обновления данных</li> <li>• S7-300: <ul style="list-style-type: none"> <li>- максимальное число параллельных заданий/экземпляров превышено</li> <li>- экземпляры DB были перезагружены при CPU-RUN</li> <li>- возможно при первом вызове</li> </ul> </li> </ul>

### Консистентность данных

#### S7-300:

Чтобы гарантировать консистентность данных, Вы вновь можете записывать в область передачи SD\_1, используемую в текущий момент времени, только после того, как операция передачи завершится. А это будет иметь место только тогда, когда значение параметра состояния DONE станет равным 1.

#### S7-400:

Когда активирована передача данных (по переднему фронту сигнала на входе REQ) данные из области для передачи SD\_i копируются из программы пользователя. После вызова SFB Вы можете записывать в эту область без риска нарушения передаваемых данных.

---

#### Примечание

Передача данных успешно завершается только тогда, если параметр состояния DONE = 1.

---

## 19.5 Некоординированный прием данных с помощью SFB9/ FB9 "URCV"

### Описание

SFB9/ FB9 "URCV" асинхронно принимает данные от SFB/FB типа "USEND" удаленного партнера по связи и копирует их в сконфигурированную область памяти для приема данных.

S7-300: Параметры R\_ID, ID и RD\_1 обновляются при каждом нарастающем фронте сигнала на входе EN\_R. После выполнения задания Вы можете назначить новые значения для параметров R\_ID, ID и RD\_1.

S7-400: Области принимаемых данных указываются параметрами RD\_1 – RD\_4.

Однако Вам нужно убедиться в том, что области, определенные параметрами RD\_i/ RD\_1 и SD\_i/ SD\_1 (SD\_i принадлежит соответствующему SFB/FB "USEND" партнера по связи) совпадают друг с другом

- по числу,
- по размеру,
- по типу данных.

Параметр R\_ID должен быть одинаковым для обоих SFB/FB.

Успешное завершение передачи данных индицируется единичным значением параметра состояния NDR: NDR = 1.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разрешение на прием". Устройство готово к приему данных при установленном входе.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
R_ID	INPUT	DWORD	I, Q, M, D, L, константа	Параметр адресации R_ID. См. "Общие параметры SFB и SFC для S7-соединения".
NDR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния NDR: 0: задание не началось или еще продолжается; 1: данные приняты без ошибок.

Параметр	Описание	Тип данных	Область памяти	Характеристика
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
S7-300: RD_1  S7-400: RD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	M, D, T, Z  I, Q, M, D, T, C	Указатель на i-ю область принимаемых данных. Разрешенные типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER Если параметр ANY указывает на DB, то последний должен быть определен (например, P# DB10.DBX5.0 Byte 10).

### Информация об ошибках

ERROR	STATUS (десятичное число)	Объяснение
0	9	Предупреждение о потере информации: более старые принятые данные заменяются более новыми принятыми данными.
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное)</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP)</li> <li>соединение с партнером еще не установлено.</li> </ul>
1	4	Ошибки в указателях областей приема RD_i, включая длину данных или тип данных.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>был задан экземпляр DB, который не принадлежит SFB9</li> <li>был задан не экземпляр DB, а общедоступный DB</li> <li>экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>

ERROR	STATUS (десятичное число)	Объяснение
1	18	R_ID уже существует в идентификаторе (ID) соединения.
1	19	Соответствующий SFB/FB "USEND" передает данные быстрее, чем они могут копироваться в области приема посредством SFB/FB "URCV".
1	20	Не хватает доступной рабочей памяти. <ul style="list-style-type: none"> <li>• H-System: первый вызов SFB во время обновления данных</li> <li>• S7-300: <ul style="list-style-type: none"> <li>- максимальное число параллельных заданий/экземпляров превышено</li> <li>- экземпляры DB были перезагружены при CPU-RUN</li> <li>- возможно при первом вызове</li> </ul> </li> </ul>

### Консистентность данных

Данные принимаются согласованно, если Вы помните следующие указания:

- S7-300: После того как значение параметра состояния NDR станет равным 1, Вы должны немедленно вызвать SFB9 "URCV" со значением 0 в EN\_R. Это гарантирует, что область приема не будет перезаписана вновь прежде, чем Вы ее проверите.
- Полностью проверьте использованные в последний раз области приема (RD\_1) прежде, чем Вы освободите блок для нового приема (вызов со значением 1 на управляющем входе EN\_R).

S7-400: После того как значение параметра состояния NDR принимает значение 1, новые полученные данные поступили в области приема (RD\_i). Новый вызов блока может вызвать замену этих данных (перезапись) в этих областях новыми принятыми данными. Для предотвращения этого Вы должны вызывать SFB9 "URCV" (как при циклической обработке блока) со значением 0 в EN\_R, пока не завершена обработка принятых данных.

## 19.6 Передача сегментированных данных с помощью SFB/FB12 "BSEND"

### Описание

SFB/FB 12 "BSEND" передает данные блоку SFB/FB типа "BRCV" удаленного партнера. При этом типе передачи данных между коммуникационными партнерами может транспортироваться наибольшее количество данных, в сравнении со всеми другими коммуникационными SFB/FB для сконфигурированных S7-соединений, а именно 32768 байтов для S7-300 и 65534 байтов для S7-400.

Причина этого заключается в том, что область передаваемых данных, сегментирована. Каждый сегмент передается партнеру по связи отдельно. Получение последнего сегмента получатель подтверждает партнеру независимо от вызова соответствующего блока SFB/FB "BRCV".

S7-300: Задание на передачу активируется после вызова блока, когда имеется передний фронт сигнала на управляющем входе REQ. Параметры R\_ID, ID, SD\_1 и LEN пересылаются при каждом нарастающем фронте сигнала на управляющем входе REQ. После завершения задания Вы можете присвоить новые значения параметрам R\_ID, ID, SD\_1 и LEN. Для передачи сегментированных данных блок должен периодически вызываться в пользовательской программе. Стартовый адрес и максимальная длина данных для передачи задается параметром SD\_1. Вы можете определять длину поля данных для задания на пересылку данных с помощью параметра LEN.

S7-400: Задание на передачу активируется после вызова блока, когда имеется нарастающий фронт сигнала на управляющем входе REQ. Передача данных из памяти пользователя асинхронна по отношению к обработке программы пользователя. Стартовый адрес и максимальная длина данных для передачи задается параметром SD\_1. Вы можете определять длину поля данных для задания на пересылку данных с помощью параметра LEN. В данном случае значение параметра LEN заменяет значение длины из параметра SD\_1.

Параметр R\_ID должен быть идентичен для двух соответствующих блоков SFB/FB.

Если имеется нарастающий фронт сигнала на управляющем входе R, текущая передача данных отменяется.

Успешное завершение передачи отображается значением 1 в параметре состояния DONE: DONE = 1.

Новое задание на передачу данных не может быть выполнено, пока предыдущее задание не завершено, т. е. пока параметры DONE или ERROR не примут значение 1.

Благодаря асинхронной передаче данных, новая передача может начаться только, если ранее переданные данные были восстановлены при вызове SFB/FB партнера.

Пока данные не восстановлены, значение состояния 7 (status value 7) (см. ниже) будет выдаваться при вызове SFB/FB "BSEND".

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данных при появлении фронта сигнала.
R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления «сброс». Активирует отмену обмена данных при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
R_ID	INPUT	DWORD	I, Q, M, D, L, константа	Параметр адресации R_ID. Сч.общие параметры SFB/FB и SFC/FC для S7-соединения. В случае подключения через CP 441 к S5 или дополнительным устройствам, R_ID содержит информацию об адресе удаленного устройства. За дополнительной информацией обратитесь к описанию CP 441.
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок

Параметр	Описание	Тип данных	Область памяти	Характеристика
SD_1	IN_OUT	ANY	S7-300: M, D  S7-400: I, Q, M, D, T, Z	Указатель на область передачи. Разрешенные типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, R# DB10.DBX5.0 Byte 10).
LEN	IN_OUT	WORD	I, Q, M, D, L	Длина массива передаваемых данных в байтах.



**Информация об ошибках**

Следующая таблица содержит всю относящуюся к SFB/FB 12 информацию об ошибках, которая может выводиться в параметрах ERROR и STATUS.

<b>ERROR</b>	<b>STATUS (десятичное число)</b>	<b>Объяснение</b>
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	<ul style="list-style-type: none"> <li>• Ошибка связи, например, не загружено описание соединения (локальное или удаленное);</li> <li>• Разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP);</li> <li>• Соединение с партнером по связи не установлено.</li> </ul>
1	2	Отрицательное подтверждение от SFB партнера. Функция не может быть выполнена.
1	3	R_ID неизвестен в соединении, задаваемом ID, или еще не был вызван блок для приема.
1	4	Ошибка в указателе области передачи SD_1, включая длину данных или тип данных, либо с LEN было передано значение 0.
1	5	Был выполнен запрос на сброс.
1	6	SFB/FB партнера находится в состоянии DISABLED [заблокирован] (EN_R имеет значение 0). Проверьте также входные параметры блока BRCV на консистентность с блоком BSEND.
1	7	SFB/FB партнера находится в состоянии ошибки (блок приема не вызывался вновь после последней передачи данных).
1	8	Отказано в доступе к удаленному объекту в памяти пользователя: область назначения для соответствующего SFB/FB 13 "BRCV" слишком мала. Соответствующий SFB/FB 13 "BRCV" выдает ERROR = 1, STATUS = 4.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>• был задан экземпляр DB, который не принадлежит SFB12</li> <li>• был задан не экземпляр DB, а общедоступный DB</li> <li>• экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>
1	18	R_ID уже существует в идентификаторе (ID) соединения.
1	20	Не хватает доступной рабочей памяти. <ul style="list-style-type: none"> <li>• H-System: первый вызов SFB во время обновления данных</li> <li>• S7-300:               <ul style="list-style-type: none"> <li>- максимальное число параллельных заданий/экземпляров превышено</li> <li>- экземпляры DB были перезагружены при CPU-RUN</li> <li>- возможно при первом вызове</li> </ul> </li> </ul>

### **Консистентность данных**

Чтобы гарантировать консистентность данных, Вы можете записывать вновь в область передачи SD\_i, используемую в текущий момент времени, только после того, как операция передачи завершится. Это будет иметь место тогда, когда значение параметра состояния DONE станет равным 1.

## 19.7 Прием сегментированных данных с помощью SFB/FB13 "BRCV"

### Описание

SFB/FB 13 "BRCV" принимает данные от SFB/FB типа "BSEND" удаленного партнера по связи. После приема каждого сегмента данных в SFB/FB партнера передается подтверждение и обновляется параметр LEN.

После того, как блок был вызван и на управляющий вход EN\_R подано значение 1, блок готов принимать данные. Активированное задание может быть снято путем задания значения 0 параметру EN\_R: EN\_R = 0.

Начальный адрес и максимальный размер области для приема определяются в параметре RD\_1. Размер (длина) последовательности принятых данных отображается в параметре LEN.

S7-300: Параметры R\_ID, ID, RD\_1 обновляются, когда появляется передний фронт сигнала на управляющем входе EN\_R. После завершения задания Вы можете присвоить новые значения параметрам R\_ID, ID, RD\_1.

Для передачи сегментированных данных блок должен периодически вызываться в пользовательской программе.

S7-400: Прием данных из памяти пользователя выполняется асинхронно по отношению к обработке программы пользователя.

Параметр R\_ID должен быть идентичен для двух соответствующих блоков SFB/FB.

Безошибочное завершение приема всех сегментов данных отображается значением 1 в параметре состояния NDR: NDR = 1. Принятые данные остаются неизменными, пока повторно вызывается SFB/FB 13 со единичным значением параметра EN\_R: EN\_R = 1.

Если блок вызывается во время асинхронного приема данных, то это приводит к выводу предупреждения в параметре состояния STATUS; если вызов блока выполняется в то время, когда 0 приложен к управляющему входу EN\_R, то прием данных прекращается и SFB/FB приходит в свое исходное состояние.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разрешение на прием", сигнализирующий о готовности блока к приему данных, когда вход установлен.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
R_ID	INPUT	DWORD	I, Q, M, D, L, константа	Параметр адресации R_ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения". В случае подключения через CP 441 к S5 или дополнительным устройствам, R_ID содержит информацию об адресе удаленного устройства. За дополнительной информацией обратитесь к описанию CP 441.
NDR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния NDR: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
RD_1	IN_OUT	ANY	S7-300: M, D  S7-400: I, Q, M, D, T, C	Указатель на область приема. Информация о длине указывает максимальный размер блока принимаемых данных. Разрешенные типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, R# DB10.DBX5.0 Byte 10).
LEN	IN_OUT	WORD	I, Q, M, D, L	Длина уже принятых данных, в байтах.

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB13 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	17	Предупреждение: блок, принимающий данные асинхронно.
0	25	Идет прием данных. Параметр LEN содержит величину уже принятых данных.
1	1	<ul style="list-style-type: none"> <li>• Ошибки связи, например, не загружено описание соединения (локальное или удаленное)</li> <li>• Разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP);</li> <li>• Соединение с партнером по связи еще не установлено.</li> </ul>
1	2	Функция не может быть выполнена.
1	4	Ошибка в указателе области приема RD_1, включая длину данных или тип данных (переданный блок данных длиннее, чем область приема).
1	5	Принят запрос на сброс, передача не завершена.
1	8	Ошибка доступа в соответствующем блоке SFB/FB 12 "BSEND". После последнего корректно принятого сегмента ERROR = 1 и STATUS = 4.
1	10	Доступ к локальной памяти пользователя невозможен (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>• был задан экземпляр DB, который не принадлежит SFB13</li> <li>• был задан не экземпляр DB, а общедоступный DB</li> <li>• экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>
1	18	R_ID уже существует в идентификаторе (ID) соединения.
1	20	Не хватает доступной рабочей памяти. <ul style="list-style-type: none"> <li>• H-System: первый вызов SFB во время обновления данных</li> <li>• S7-300: <ul style="list-style-type: none"> <li>- максимальное число параллельных заданий/экземпляров превышено</li> <li>- экземпляры DB были перезагружены при CPU-RUN</li> <li>- возможно при первом вызове</li> </ul> </li> </ul>

### Консистентность данных

Данные принимаются согласованно, если Вы помните следующие указания:

- Проверьте использованные в последний раз области приема (RD\_1) прежде, чем Вы вызовете блок для нового приема (вызов со значением 1 на управляющем входе EN\_R).

### **Особый случай приема данных (только для S7-400)**

Если приемный CPU с блоком BRCV, готовый к приему данных (т.е. уже был сделан вызов с параметром EN\_R = 1), переходит в режим STOP до того, как соответствующий блок передачи пошлет первый сегмент данных при выполнении задания, произойдет следующее:

- ранее принятые данные, после приема которых CPU переходит в STOP-режим, полностью вводятся в область назначения (приема данных);
- SFB "BSEND" партнера по связи принимает положительное подтверждение;
- любое дополнительное задание "BSEND" не может больше приниматься CPU, находящимся в STOP-режиме;
- пока CPU находится в STOP-режиме, значения параметров NDR и LEN имеют значение 0.

Чтобы не потерять информацию принимаемых данных, Вы должны выполнить горячий рестарт CPU-получателя и вызывать SFB 13 "BRCV" с параметром EN\_R = 1.

## 19.8 Запись данных в удаленный CPU с помощью SFB/FB15 "PUT"

### Описание

С помощью SFB15 "PUT" Вы можете записывать данные в удаленный CPU.

S7-300: Данные посылаются, когда появляется передний фронт сигнала на управляющем входе REQ. После завершения задания Вы можете присвоить новые значения параметрам ID, ADDR\_1 и SD\_1.

S7-400: SFB запускается, когда появляется передний фронт сигнала на управляющем входе REQ. В процессе передачи указатели на области назначения для записи (ADDR\_i) и данные (SD\_i) посылаются CPU партнера по связи.

Удаленный партнер сохраняет требуемые данные по адресам, предоставленным вместе с данными, и возвращает подтверждение выполнения.

Если ошибок не было, то это отображается значением 1 в параметре состояния DONE при следующем вызове SFB/FB.

Задание на запись может быть активировано вновь только после того, как завершится последнее задание.

Удаленный CPU может находиться в режиме RUN или STOP.

Ошибки и предупреждения выводятся с помощью параметров ERROR и STATUS, если происходят ошибки доступа во время записи данных или выполнение проверки приводит к ошибке.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данных при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения.
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.

ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	<p>Параметры состояния ERROR и STATUS отображают:</p> <p>При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок &lt;&gt; 0000H: Предупреждение. STATUS содержит детальную информацию</p> <p>При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок</p>
S7-300: ADDR_1	IN_OUT	ANY	M, D	Указатели на области в CPU партнера, в которые будут записываться данные.
S7-400: ADDR_I (1≤i≤4)				
S7-300: SD_1	IN_OUT	ANY	S7-300: M, D	<p>Указатели на области в локальном CPU, содержащие данные, которые должны передаваться.</p> <p>Разрешены типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.</p> <p>Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, R# DB10.DBX5.0 Byte 10).</p>
S7-400: SD_i (1≤i≤4)			S7-400 I, Q, M, D, T, C	

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB/FB15 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	<p>Проблемы связи, например:</p> <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное);</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP);</li> <li>Соединение с партнером по связи пока не установлено.</li> </ul>



ERROR	STATUS (десятичное число)	Объяснение
1	2	Отрицательное подтверждение от устройства партнера. Функция не может быть выполнена.
1	4	Ошибки в указателях областей передачи SD_i, включая длину данных или тип данных.
1	8	Ошибка доступа в CPU партнера.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>• был задан экземпляр DB, который не принадлежит SFB15</li> <li>• был задан не экземпляр DB, а общедоступный DB</li> <li>• экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>
1	20	Не хватает доступной рабочей памяти. <ul style="list-style-type: none"> <li>• H-System: первый вызов SFB во время обновления данных</li> <li>• S7-300: <ul style="list-style-type: none"> <li>- максимальное число параллельных заданий/экземпляров превышено</li> <li>- экземпляры DB были перезагружены при CPU-RUN</li> <li>- возможно при первом вызове</li> </ul> </li> </ul>

### Консистентность данных для S7-300

Чтобы гарантировать консистентность данных, Вы можете использовать вновь область передачи SD\_i, используемую в текущий момент времени, только после того, как операция передачи завершится. Это будет иметь место тогда, когда значение параметра состояния DONE станет равным 1.

### Консистентность данных для S7-400

При активном процессе пересылки данных, передаваемые данные из области передачи SD\_i копируются из пользовательской программы. После вызова блока Вы можете записывать в эти области информацию без нарушения текущего процесса передачи данных.

---

#### Примечание

Операция передачи данных завершается только тогда, когда параметр состояния DONE принимает значение 1.

---

## 19.9 Чтение данных из удаленного CPU с помощью SFB/FB14 "GET"

### Описание

С помощью SFB/FB14 "GET" Вы можете читать данные из удаленного CPU.

S7-300: Данные считываются, когда появляется передний фронт сигнала на управляющем входе REQ. Параметры ID, ADDR\_1 и RD\_1 пересылаются при каждом появлении переднего фронта сигнала на управляющем входе REQ. После завершения задания Вы можете присваивать новые значения параметрам ID, ADDR\_1 и RD\_1.

S7-400: SFB запускается, когда появляется передний фронт сигнала на управляющем входе REQ. В процессе передачи указатели на области считывания (ADDR\_i) посылаются CPU партнера по связи.

Удаленный партнер возвращает данные.

Принятые данные копируются в сконфигурированные области приема (RD\_i) при следующем вызове SFB/FB.

Удостоверьтесь, что области, определенные параметрами ADDR\_i и RD\_i ( $1 \leq i \leq 4$ ), совпадают по длине и типу данных.

Завершение задания отображается значением 1 в параметре состояния NDR.

Задание на чтение может быть активировано вновь только после того, как завершится предыдущее задание.

Удаленный CPU может находиться в режимах RUN или STOP.

Ошибки и предупреждения выводятся с помощью параметров ERROR и STATUS, если происходят ошибки доступа во время считывания данных или выполнение проверки приводит к ошибке.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данных при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения.
NDR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния NDR: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.

ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	<p>Параметры состояния ERROR и STATUS отображают:</p> <p>При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок &lt;&gt; 0000H: Предупреждение. STATUS содержит детальную информацию</p> <p>При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок</p>
S7-300: ADDR_1	IN_OUT	ANY	S7-300: M, D	Указатели на области в CPU партнера, которые будут считываться.
S7-400: ADDR_I (1≤i≤4)			S7-400: I, Q, M, D, T, C	
S7-300: RD_1	IN_OUT	ANY	S7-300: M, D	<p>Указатели на области в локальном CPU, содержащие данные, которые должны считываться.</p> <p>Разрешены типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.</p> <p>Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, P# DB10.DBX5.0 Byte 10).</p>
S7-400: RD_i (1≤i≤4)			S7-400: I, Q, M, D, T, C	

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB14 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.

ERROR	STATUS (десятичное число)	Объяснение
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>• не загружено описание соединения (локальное или удаленное);</li> <li>• разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP);</li> <li>• Соединение с партнером по связи пока не установлено.</li> </ul>
1	2	Отрицательное подтверждение от устройства партнера. Функция не может быть выполнена.
1	4	Ошибки в указателях областей приема RD_i, включая длину данных или тип данных.
1	8	Ошибка доступа в CPU партнера.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>• был задан экземпляр DB, который не принадлежит SFB14</li> <li>• был задан не экземпляр DB, а общедоступный DB</li> <li>• экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>
1	20	Не хватает доступной рабочей памяти. <ul style="list-style-type: none"> <li>• H-System: первый вызов SFB во время обновления данных</li> <li>• S7-300: <ul style="list-style-type: none"> <li>- максимальное число параллельных заданий/экземпляров превышено</li> <li>- экземпляры DB были перезагружены при CPU-RUN</li> <li>- возможно при первом вызове</li> </ul> </li> </ul>

### Консистентность данных

Данные консистентны, если Вы придерживаетесь следующего правила:

Проверяйте заполненность области для приема данных RD\_i перед тем, как запустить новое задание.

## 19.10 Передача данных на принтер с помощью SFB16 "PRINT"

### Описание

SFB16 "PRINT" передает данные и команду форматирования на удаленный принтер, например, через CP 441.

При нарастающем фронте сигнала на управляющем входе REQ описание формата (FORMAT) и данные (SD\_i) передаются на принтер, выбранный посредством ID и PRN\_NR.

Если Вы используете не все четыре области передачи, то Вы должны убедиться в том, что первая область описана параметром SD\_1, вторая область (если она существует) – параметром SD\_2, третья область (если она существует) – параметром SD\_3.

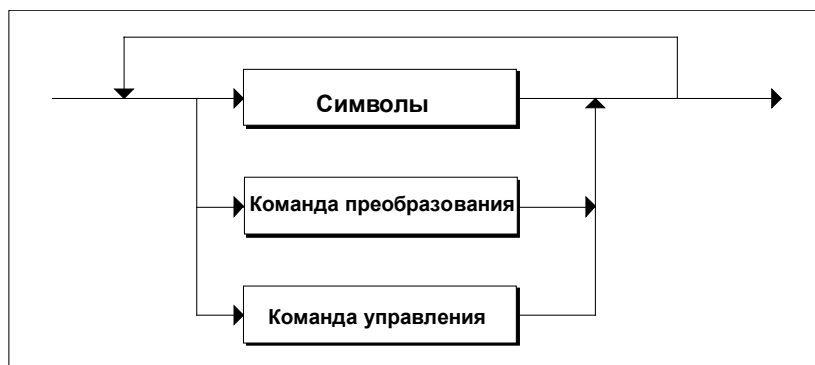
Успешное выполнение задания отображается параметром состояния DONE, ошибки отображаются параметрами состояния ERROR и STATUS.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данных при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
PRN_NR	IN_OUT	BYTE	I, Q, M, D, L	Номер принтера.
FORMAT	IN_OUT	STRING	I, Q, M, D, L	Описание формата.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SD_i (1 ≤ i ≤ 4)	IN_OUT	ANY	M, D, T, C	Указатель на i-ю область передаваемых данных. Разрешены типы данных BOOL (не разрешен битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, P# DB10.DBX5.0 Byte 10).

### Проходной (in/out) параметр FORMAT

Символьная строка FORMAT содержит печатные символы и элементы формата. Она имеет следующую структуру:

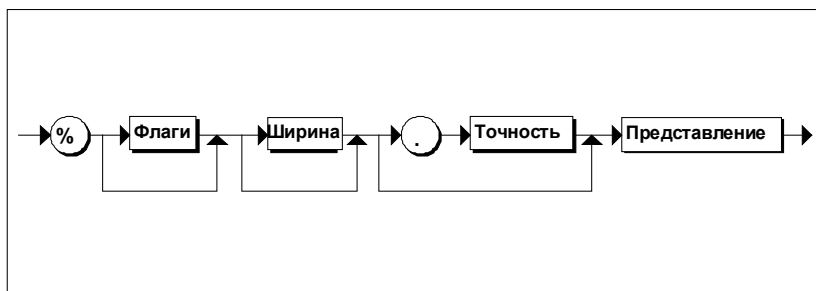


Для каждой области, передаваемой для печати (с SD\_1 по SD\_4), должна иметься одна команда преобразования в параметре FORMAT. Команды преобразования применяются к областям передачи (SD\_i) в том порядке, в котором они формулируются. Символы и команды могут следовать за друг другом в любом порядке.

- Символы

Разрешены следующие символы:

- все печатные символы
- \$\$ (символ доллара), \$' (отдельный апостроф), \$L и \$l (перевод строки), \$P и \$p (страница), \$R и \$r (возврат каретки), \$T и \$t (табулятор).



Синтаксическая диаграмма команды преобразования

Элемент команды преобразования	Значение
Флаги	<ul style="list-style-type: none"> <li>отсутствуют : выровненный по правому краю вывод.</li> <li>- : выровненный по левому краю вывод .</li> </ul>
Ширина	<ul style="list-style-type: none"> <li>отсутствует : вывод в стандартном представлении.</li> <li>n : выводятся точно n символов. Если вывод выровнен по правому краю, то ему могут предшествовать пробелы; если вывод выровнен по левому краю, то пробелы следуют после символов.</li> </ul>
Точность	<p>Точность имеет значение только для представлений A, D, F и R (см. следующую таблицу).</p> <ul style="list-style-type: none"> <li>отсутствует : вывод в стандартном представлении.</li> <li>0 : без вывода десятичной точки или десятичных разрядов в представлениях F и R.</li> <li>n <ul style="list-style-type: none"> <li>в случае F и R: вывод десятичной точки и n десятичных разрядов</li> <li>в случае A и D (дата): количество цифр для года: возможны значения 2 и 4.</li> </ul> </li> </ul>
Представление	<p>Следующая таблица содержит:</p> <ul style="list-style-type: none"> <li>возможные представления</li> <li>типы данных, возможные для каждого представления</li> <li>стандартный формат для каждого представления (распечатка производится в стандартном представлении, если в параметре FORMAT не заданы ширина и точность) и их максимальная длина.</li> </ul>

Следующая таблица показывает возможные виды представления в команде преобразования параметра FORMAT.

Представление	Возможные типы данных	Стандартное представление		Комментарий
		Пример	Длина	
A, a	DATE	25.07.1996	10	-
	DWORD			
C, c	CHAR	K	1	-
	BYTE	M	1	
	WORD	KL	2	
	DWORD	KLMN	4	

Представление	Возможные типы данных	Стандартное представление		Комментарий
			Число символов	
D, d	ARRAY of CHAR	KLMNOP		-
	ARRAY of BYTE			
F, f	DATE	1996-07-25	10	-
	DWORD			
H, h	REAL	0.345678	8	Шестнадцатеричное представление
	DWORD			
I, i	Все типы данных, включая ARRAY of BYTE (байтовый массив)	В зависимости от типа данных	В зависимости от типа данных	-
	INT	- 32 768	макс. 6	
N, n	WORD	- 2 147 483 648	макс. 11	Соответствующая область передачи SD_i содержит ссылку (в виде номера) на текст, который нужно напечатать. Текст находится в модуле (например, CP 441), который создает печатаемую строку. Если текст под указанным номером не найден, то выводится *****.
	WORD	Вывод текста	-	
R, r	REAL	0.12E-04	8	-
	DWORD			
S, s	STRING	Вывод текста		-
T, t	TIME	2d_3h_10m_5s_250ms	макс. 21	Если появляется ошибка, то выводится *****.
	DWORD			
U, u	BYTE	255	макс. 3	-
	WORD	65 535	макс. 5	
	DWORD	4 294 967 295	макс. 10	
X, x	BOOL	1	1	-
	BYTE	101 ..	8	
	WORD	101 ..	16	
	DWORD	101 ..	32	
Z, z	TIME_OF_DAY	15:38:59.874	12	-

В пунктах этой таблицы, в которых указана максимальная длина для представления, фактическая длина, конечно, может быть короче.



**Примечание**

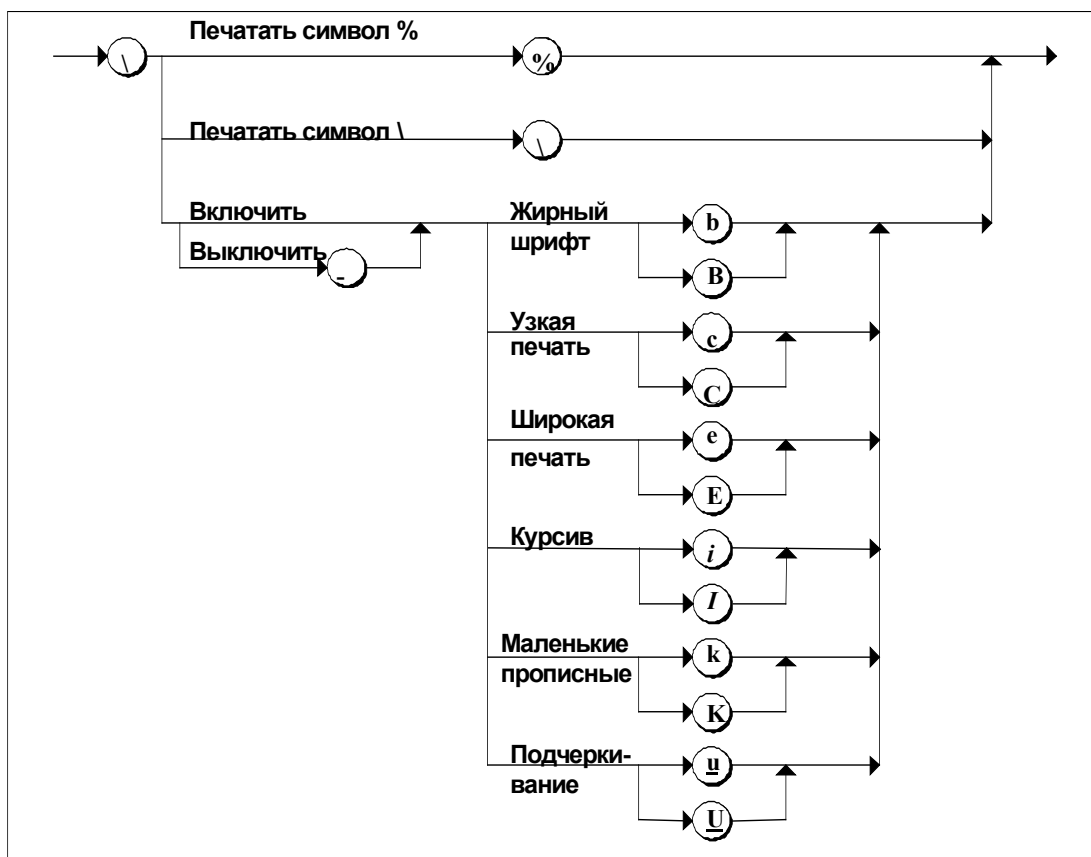
Для типов данных C и S следующие пункты зависят от используемого принтера:

- какие символы могут печататься
- что печатает принтер вместо непечатаемых символов, если драйвер принтера не имеет таблицы преобразования этих символов.

Команда управления

С помощью команды управления Вы можете делать следующее:

- печатать символы % и \
- изменять параметры настройки принтера.



Синтаксическая диаграмма команды управления

Если Вы пытаетесь, например, заблокировать неразрешенный шрифт или выполнить функцию, которую принтер не распознает, то команда управления игнорируется. Следующая таблица содержит ошибки, которые могут происходить в связи с проходным параметром FORMAT.

Ошибка	Вывод принтера
Команда преобразования не может быть выполнена.	Выводятся символы * в соответствии с (максимальной) длиной заданного по умолчанию представления или заданной шириной.
Заданная ширина слишком мала.	В представлениях A, C, D, N, S, T и Z печатается столько символов, сколько задано посредством выбранной ширины. При всех других представлениях на протяжении заданной ширины печатаются символы *.
Слишком много команд преобразования.	Команды преобразования, для которых нет указателя области передачи SD <sub>i</sub> , игнорируются.
Слишком мало команд преобразования.	Области передачи, для которых нет команды преобразования, не распечатываются.
Неопределенные или неподдерживаемые команды преобразования.	Печатается .....
Неполная команда преобразования.	Печатается .....
Неопределенные или неподдерживаемые команды управления.	Команды управления, не соблюдающие синтаксис, показанный на вышеприведенном рисунке, игнорируются.

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB16 "PRINT" информацию об ошибках, которая может быть распечатана с помощью параметров ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не активно, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное)</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP).</li> </ul>
1	2	Отрицательное подтверждение от принтера. Функция не может быть выполнена.
1	3	PRN NR неизвестен в канале связи, заданном посредством ID.
1	4	Ошибка в проходном (in/out) параметре FORMAT или в указателях области передачи SD <sub>i</sub> относительно длины данных или типа данных.
1	6	Удаленный принтер находится в режиме OFFLINE.
1	7	Удаленный принтер не готов к работе (например, нет бумаги).
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	13	Ошибка в проходном (in/out) параметре FORMAT.
1	20	Недостаточно доступной рабочей памяти. H-System: вызов SFB во время обновления данных.

### Количество передаваемых данных

Количество данных, которые можно передать удаленному принтеру, не должно превышать максимальную длину.

Эта максимальная длина данных рассчитывается следующим образом:

максимальная длина = 420 – формат

Формат – это текущая длина параметра FORMAT в байтах. Данные, подлежащие распечатке, могут быть распределены по одной или большему количеству областей передачи.

## 19.11 Инициализация теплого или холодного рестарта в удаленном устройстве с помощью SFB19 "START"

### Описание

SFB19 "START" при нарастающем фронте сигнала на управляющем входе REQ активирует теплый или холодный рестарт в удаленном устройстве, адресованном посредством ID. Если удаленная система отказоустойчива, то запрос рестарта имеет влияние на один или все CPU этой системы – в зависимости от значения PI\_NAME. Если удаленным устройством является CPU, то должны быть выполнены следующие условия:

- CPU находится в состоянии STOP
- переключатель CPU установлен в положение "RUN" или "RUN-P".

Как только теплый или холодный рестарт завершается, устройство переключается в режим RUN и передает положительное подтверждение выполнения. Когда положительное подтверждение оценено, параметр состояния DONE устанавливается в 1. Если происходят какие-либо ошибки, то они отображаются параметрами состояния ERROR и STATUS.

Дальнейший теплый или холодный рестарт в том же самом удаленном устройстве может быть активирован только после того, как завершится последний полный рестарт.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данными при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок

Параметр	Описание	Тип данных	Область памяти	Характеристика
PI_NAME	IN_OUT	ANY	I, Q, M, D, T, C	Указатель на область памяти, в которой расположено имя запускаемой программы (код ASCII). Это имя не должно содержать более 32 символов. В PLC S7 оно должно быть P_PROGRAM. Для H-систем возможны следующие имена <ul style="list-style-type: none"> <li>• P_PROGRAM (запуск задания для всех CPU в H- системе)</li> <li>• P_PROG_0 (запуск задания для CPU в стойке 0 в H-системе)</li> <li>• P_PROG_1 (запуск задания для CPU в стойке 1 в H-системе)</li> </ul>
ARG	IN_OUT	ANY	I, Q, M, D, T, C	Параметр выполнения: <ul style="list-style-type: none"> <li>• Если Вы не задаете значение для ARG, то в удаленном устройстве выполняется теплый рестарт.</li> <li>• Если Вы задаете значение "C", то в удаленном устройстве выполняется холодный рестарт (если удаленное устройство способно к запуску этого типа).</li> </ul>
IO_STATE	IN_OUT	BYTE	I, Q, M, D, L	В настоящее время не имеет значения. Не присваивайте значение этому параметру, если Вашим партнером по связи является программируемый контроллер S7.

**Информация об ошибках**

Следующая таблица содержит всю относящуюся к SFB19 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное)</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP).</li> </ul>
1	2	Отрицательное подтверждение от устройства партнера. Функция не может быть выполнена.
1	3	Имя программы, введенное для PI_NAME, неизвестно.
1	4	Ошибка в указателях PI_NAME или ARG, включая длину данных или тип данных.
1	7	Полный рестарт в устройстве партнера невозможен.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>был задан экземпляр DB, который не принадлежит SFB19</li> <li>был задан не экземпляр DB, а общедоступный DB</li> <li>экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>
1	20	Недостаточно доступной рабочей памяти. H-System: вызов SFB во время обновления данных.

## 19.12 Переключение удаленного устройства в состояние STOP с помощью SFB20 "STOP"

### Описание

SFB20 "STOP" при нарастающем фронте сигнала на управляющем входе REQ активирует переключение в режим STOP в удаленном устройстве, адресованном посредством ID. Переключение режима возможно, когда устройство находится в режиме RUN, HOLD или STARTUP.

Если удаленная система отказоустойчива, то запрос рестарта удаленная система отказоустойчива, то запрос рестарта имеет влияние на один или все CPU этой системы – в зависимости от значения PI\_NAME.

Успешное выполнение задания отображается значением 1 в параметре состояния DONE. Если происходят какие-либо ошибки, то они отображаются параметрами состояния ERROR и STATUS.

Изменение режима в том же самом удаленном устройстве может быть запущено только тогда, когда предыдущий вызов SFB20 полностью завершен.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данными при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для ".
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок

Параметр	Описание	Тип данных	Область памяти	Характеристика
PI_NAME	IN_OUT	ANY	I, Q, M, D	Указатель на область памяти, в которой расположено имя запускаемой программы (код ASCII). Это имя не должно содержать более 32 символов. В PLC S7 оно должно быть P_PROGRAM. Для H-систем возможны следующие имена <ul style="list-style-type: none"> <li>• P_PROGRAM (запуск задания для всех CPU в H- системе)</li> <li>• P_PROG_0 (запуск задания для CPU в стойке 0 в H-системе)</li> <li>• P_PROG_1 (запуск задания для CPU в стойке 1 в H-системе)</li> </ul>
IO_STATE	IN_OUT	BYTE	I, Q, M, D, L	В настоящее время не имеет значения. Не присваивайте значение этому параметру, если Вашим партнером по связи является программируемый контроллер S7.

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB20 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>• не загружено описание соединения (локальное или удаленное)</li> <li>• разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP).</li> </ul>
1	2	Отрицательное подтверждение от устройства партнера. Функция не может быть выполнена.
1	3	Имя программы, введенное для PI_NAME, неизвестно.
1	4	Ошибка в указателе PI_NAME, включая длину данных или тип данных.
1	7	Устройство партнера уже находится в состоянии STOP.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>• был задан экземплярный DB, который не принадлежит SFB20</li> <li>• был задан не экземплярный DB, а общедоступный DB</li> <li>• экземплярный DB не был найден (загрузка нового экземплярного DB из PG).</li> </ul>
1	20	Недостаточно доступной рабочей памяти. H-System: вызов SFB во время обновления данных.

## 19.13 Инициализация горячего рестарта в удаленном устройстве с помощью SFB21 "RESUME"

### Описание

SFB21 "RESUME" при нарастающем фронте сигнала на управляющем входе REQ активирует горячий рестарт в удаленном устройстве, выбранном с помощью ID.

Если удаленным устройством является CPU, то должны быть выполнены следующие условия:

- CPU находится в состоянии STOP
- переключатель CPU установлен в положение "RUN" или "RUN-P"
- при создании конфигурации с помощью STEP 7 разрешен ручной горячий перезапуск
- не должно быть условий, препятствующих горячему рестарту.

Как только горячий рестарт завершается, устройство переходит в режим RUN и передает положительное подтверждение выполнения. Когда положительное подтверждение оценено, параметр DONE состояния устанавливается в 1. Если происходят какие-либо ошибки, то они отображаются параметрами состояния ERROR и STATUS.

Рестарт в том же самом удаленном устройстве может быть активирован вновь только после того, как завершится предыдущий горячий рестарт.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данными при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок



PI_NAME	IN_OUT	ANY	I, Q, M, D	Указатель на область памяти, в которой расположено имя запускаемой программы (код ASCII). Это имя не должно содержать более 32 символов. В ПЛК S7 это должно быть P_PROGRAM.
ARG	IN_OUT	ANY	I, Q, M, D, T, C	Параметр выполнения. В настоящее время не имеет значения. Не присваивайте значение этому параметру, если Вашим партнером по связи является программируемый контроллер S7.
IO_STATE	IN_OUT	BYTE	I, Q, M, D, L	В настоящее время не имеет значения. Не присваивайте значение этому параметру, если Вашим партнером по связи является программируемый контроллер S7.

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB21 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное)</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP).</li> </ul>
1	2	Отрицательное подтверждение от устройства партнера. Функция не может быть выполнена.
1	3	Имя программы, введенное для PI_NAME, неизвестно
1	4	Ошибка в указателях PI_NAME или ARG, включая длину данных или тип данных.
1	7	Горячий рестарт невозможен.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>был задан экземплярный DB, который не принадлежит SFB21</li> <li>был задан не экземплярный DB, а общедоступный DB</li> <li>экземплярный DB не был найден (загрузка нового экземплярного DB из PG).</li> </ul>
1	20	Недостаточно доступной рабочей памяти. H-System: вызов SFB во время обновления данных.

## 19.14 Запрос состояния удаленного партнера с помощью SFB22 "STATUS"

### Описание

Используя SFB22 "STATUS" Вы можете запросить состояние удаленного партнера по связи.

Если имеется нарастающий фронт сигнала на управляющем входе REQ, задание передается удаленному партнеру. Ответ оценивается, чтобы определить, были ли проблемы. Если ошибок не было, то принятое состояние копируется в переменные PHYS, LOG и LOCAL при следующем вызове SFB. Завершение этого задания отображается значением 1 в параметре состояния NDR.

Вы можете запрашивать состояние того же самого партнера по связи вновь только после того, как завершится последний запрос.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данными при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
NDR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния NDR: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
PHYS	IN_OUT	ANY	I, Q, M, D	Физическое состояние (минимальная длина: один байт), возможные значения: • 10H функционирование • 13H требуется техническое обслуживание.
LOG	IN_OUT	ANY	I, Q, M, D	Логическое состояние (минимальная длина: один байт), возможное значение: • 00H разрешено изменение состояния.

Параметр	Описание	Тип данных	Область памяти	Характеристика
LOCAL	IN_OUT	ANY	I, Q, M, D	Состояние, если устройством партнера является CPU S7 (минимальная длина: два байта).

### Проходной (in/out) параметр LOCAL

Если партнером по связи является CPU S7, то проходной (in/out) параметр LOCAL содержит его текущее состояние. Первый байт зарезервирован, второй байт содержит ID состояния.

Режим работы	Соответствующий идентификатор
STOP (Стоп)	00H
Теплый рестарт	01H
RUN (Выполнение)	02H
Горячий рестарт	03H
HOLD (Задержка)	04H
Холодный рестарт	06H
RUN_R	09H
LINK-UP (Установление связи)	0BH
UPDATE (Обновление)	0CH

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB22 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>не загружено описание соединения (локальное или удаленное)</li> <li>разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP).</li> </ul>
1	2	Отрицательное подтверждение от устройства партнера. Функция не может быть выполнена.
1	4	Ошибка в PHYS, LOG или LOCAL, включая длину данных или тип данных.
1	8	Обращение к удаленному объекту было отклонено.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>был задан экземплярный DB, который не принадлежит SFB22</li> <li>был задан не экземплярный DB, а общедоступный DB</li> <li>экземплярный DB не был найден (загрузка нового экземплярного DB из PG).</li> </ul>
1	20	Недостаточно доступной рабочей памяти. H-System: вызов SFB во время обновления данных.

## 19.15 Прием состояния удаленного устройства с помощью SFB23 "USTATUS"

### Описание

SFB23 "USTATUS" принимает состояние устройства удаленного партнера по связи. Партнер передает свое состояние без запроса, когда происходит его изменение, если это сконфигурировано в STEP 7.

Если при вызове SFB на управляющий вход EN\_R подается значение 1 и имеется кадр от партнера, то информация о состоянии вводится в переменные PHYS, LOG и LOCAL при следующем вызове SFB. Завершение этого задания отображается значением 1 в параметре состояния NDR.

В соединении, используемом USTATUS, должна быть разблокирована передача сообщений о рабочем состоянии.

### Примечание

Вы можете использовать только один экземпляр SFB23 на соединение.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L	Параметр управления «запрос». Активирует обмен данными при появлении фронта сигнала.
ID	INPUT	WORD	I, Q, M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
NDR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния NDR: 0: задание не началось или еще продолжается; 1: задание выполнено без ошибок.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
PHYS	IN_OUT	ANY	I, Q, M, D	Физическое состояние (минимальная длина: один байт), возможные значения: • 10H функционирование • 13H требуется техническое обслуживание.

LOG	IN_OUT	ANY	I, Q, M, D	Логическое состояние (минимальная длина: один байт), возможное значение: <ul style="list-style-type: none"> <li>• 00H разрешено изменение состояния.</li> </ul>
LOCAL	IN_OUT	ANY	I, Q, M, D	Состояние, если устройством партнера является CPU S7 (минимальная длина: один байт).

### Проходной (in/out) параметр LOCAL

Если партнером по связи является CPU S7, то проходной (in/out) параметр LOCAL содержит его текущее состояние. Первый байт зарезервирован, второй байт содержит ID состояния.

Режим работы	Соответствующий идентификатор
STOP (Стоп)	00H
Теплый рестарт	01H
RUN (Выполнение)	02H
Горячий рестарт	03H
HOLD (Задержка)	04H
Холодный рестарт	06H
RUN R	09H
LINK-UP (Установление связи)	0BH
UPDATE (Обновление)	0CH

### Информация об ошибках

Следующая таблица содержит всю относящуюся к SFB23 информацию об ошибках, которая может выводиться параметрами ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	9	Предупреждение о потере информации: Более старое состояние устройства было заменено более новым состоянием устройства.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи, например: <ul style="list-style-type: none"> <li>• не загружено описание соединения (локальное или удаленное)</li> <li>• разрыв соединения (например, кабель, CPU выключен, CP в режиме STOP).</li> </ul>
1	4	Ошибка в PHYS, LOG или LOCAL, включая длину данных или тип данных.
1	10	Обращение к локальной памяти пользователя невозможно (например, обращение к DB, который был удален).
1	12	При вызове SFB <ul style="list-style-type: none"> <li>• был задан экземплярный DB, который не принадлежит SFB23</li> <li>• был задан не экземплярный DB, а общедоступный DB</li> <li>• экземпляр DB не был найден (загрузка нового экземпляра DB из PG).</li> </ul>

1	18	Для соединения, идентифицируемого ID, уже имеется экземпляр SFB23 "USTATUS".
1	19	Удаленный CPU передает данные быстрее, чем они могут приниматься SFB в программе пользователя.
1	20	Недостаточно доступной рабочей памяти. H-System: вызов SFB во время обновления данных.

## 19.16 Запрос состояния соединения, относящегося к экземпляру коммуникационного SFB, с помощью SFC62 "CONTROL"

### Описание

С помощью SFC62 "CONTROL" Вы можете запрашивать состояние соединения, относящегося к экземпляру локального коммуникационного SFB, в S7-400.

После вызова этой системной функции со значением 1 на управляющем входе EN\_R запрашивается текущее состояние соединения, относящегося к экземпляру коммуникационного SFB, выбранному с помощью I\_DB.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разрешение на прием". Сигнализирует о готовности устройства принять информацию при установленном входе.
I_DB	INPUT	BLOCK_DB	I, Q, M, D, L, константа	Номер экземпляра DB.
OFFSET	INPUT	WORD	I, Q, M, D, L, константа	Номер записи данных в мультиэкземплярном DB (если мультиэкземплярный DB не существует, то здесь нужно ввести 0).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
STATUS	OUTPUT	WORD	I, Q, M, D, L	
I_TYP	OUTPUT	BYTE	I, Q, M, D, L	Идентификатор типа блока, принадлежащего выбранному экземпляру.

Параметр	Описание	Тип данных	Область памяти	Характеристика
I_STATE	OUTPUT	BYTE	I, Q, M, D, L	<ul style="list-style-type: none"> <li>• =0: соответствующий экземпляр SFB не вызывался с момента последнего холодного/теплого рестарта или загрузки;</li> <li>• ≠0: соответствующий экземпляр SFB вызывался по крайней мере один раз с момента последнего холодного/теплого рестарта или загрузки.</li> </ul>
I_CONN	OUTPUT	BOOL	I, Q, M, D, L	Состояние соответствующего соединения; возможные значения: <ul style="list-style-type: none"> <li>• 0: соединение прервано или не установлено</li> <li>• 1: соединение существует.</li> </ul>
I_STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS запрашиваемого экземпляра коммуникационного SFB.

### Выходной параметр I\_TYP

Следующая таблица перечисляет различные типы SFB и соответствующие идентификаторы.

Тип SFB	Идентификатор (W#16#...)
USEND	00
URCV	01
BSEND	04
BRCV	05
GET	06
PUT	07
PRINT	08
START	0B
STOP	0C
RESUME	0D
STATUS	0E
USTATUS	0F
ALARM	15
ALARM_8	16
ALARM_8P	17
NOTIFY	18
AR_SEND	19
(SFB не существует; неправильный I_DB или OFFSET)	FF



**Информация об ошибках**

В SFC62 "CONTROL" выходной параметр RET\_VAL может иметь два следующих значения:

- 0000H: во время выполнения SFC ошибок не было
- 8000H: во время выполнения SFC возникла ошибка.

**Примечание**

Выходные параметры ERROR и STATUS следует проверять, даже если в выходном параметре RET\_VAL отображается значение 0000H.

ERROR	STATUS (десятичное число)	Объяснение
1	10	Обращение к локальной памяти пользователя невозможно (например, для I_TYP в качестве фактического параметра был указан байт памяти, и этот байт памяти не существует в используемом CPU).
1	12	Для номера, заданного посредством I_DB <ul style="list-style-type: none"><li>• имеется не экземплярный DB, а общедоступный DB</li><li>• нет DB, либо экземпляр разрушен.</li></ul>

## 19.17 Запрос состояния соединения с помощью FC62 "C\_CNTRL"

### Описание

С помощью FC62 "C\_CNTRL" Вы можете запрашивать состояние соединения для S7-300.

После вызова этой функции со значением 1 на управляющем входе EN\_R будет запрошено текущее состояние соединения, выбранного с помощью ID.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разрешение на прием". Готов к приему, если вход установлен.
ID	INPUT	WORD	M, D, константа	Параметр адресации ID. См. "Общие параметры SFB/FB и SFC/FC для S7-соединения".
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках.
ERROR STATUS	OUTPUT OUTPUT	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Параметры состояния ERROR и STATUS отображают: При ERROR = 0 STATUS имеет значения: 0000H: нет предупреждений и ошибок <> 0000H: Предупреждение. STATUS содержит детальную информацию При ERROR = 1 Зафиксирована ошибка. STATUS содержит детальную информацию о типе ошибок
C_CONN	OUTPUT	BOOL	I, Q, M, D, L	Состояние соответствующего соединения; возможные значения: • 0: соединение прервано или не установлено • 1: соединение существует.
C_STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS соединения: • W#16#0000: Соединение не установлено • W#16#0001: Соединение устанавливается • W#16#0002: Соединение установлено • W#16#000F: Данные о соединении недоступны (как при запуске CP) • W#16#00FF: Соединение не сконфигурировано

### Информация об ошибках

В FC62 "CC\_NTRL" выходной параметр RET\_VAL может иметь два следующих значения:

- 0000H: во время выполнения FC ошибок не было
- 8000H: во время выполнения FC возникла ошибка.

---

#### Примечание

Выходные параметры ERROR и STATUS следует проверять, даже если в выходном параметре RET\_VAL отображается значение 0000H.

---

ERROR	STATUS (десятичное число)	Объяснение
1	10	Ошибка доступа к CP. В текущий момент выполняется другое задание. Повторно иницируйте задание позднее.

## 19.18 Требования к рабочей памяти для SFB/FB системы связи S7 (S7 Communication)

Для нормальной работы SFB/FB системы связи S7 (S7 Communication) требуют определенного объема памяти для временного размещения в рабочей памяти CPU (work memory) в зависимости от пользовательских данных (область кодов). Размер требуемой памяти показан в следующей таблице:

Блок в S7-300		Требуемый размер памяти в рабочей памяти в байтах
FB8	USEND	Блок: 4583 байта, экземпляр (instance): 368 байтов
FB9	URCV	Блок: 4880 байтов, экземпляр (instance): 370 байтов
FB12	BSEND	Блок: 5284 байта, экземпляр (instance): 372 байта
FB13	BRCV	Блок: 5258 байтов, экземпляр (instance): 374 байта
FB14	GET	Блок: 4888 байтов, экземпляр (instance): 336 байтов
FB15	PUT	Блок: 4736 байтов, экземпляр (instance): 384 байта
FC 62	C_CNTRL	Блок: 546 байтов

### Замечания по поведению при прерываниях

В S7-300, SIMATIC\_NET коммуникационные блоки могут вызываться только в одном приоритетном классе.

Блок в S7-400		Требуемый размер памяти в рабочей памяти в байтах
SFB8/ SFB9	USEND/ URCV	68 + длина указанных пользовательских данных при первом вызове из SD_1,... SD_4 / RD_1,... RD_4
SFB12/ SFB13	BSEND/ BRCV	54
SFB14	GET	88 + длина указанных пользовательских данных при первом вызове из RD_1,... RD_4
SFB15	PUT	108 + длина указанных пользовательских данных при первом вызове из SD_1,... SD_4
SFB16	PRINT	78 + длина спецификации формата FORMAT + длина указанных пользовательских данных при первом вызове из SD_1,... SD_4

*(продолжение таблицы)*

Блок в S7-400		Требуемый размер памяти в рабочей памяти в байтах
SFB19	START	52 + длина указанных пользовательских данных при первом вызове из PI_NAME и ARG.
SFB20	STOP	48 + длина указанных пользовательских данных при первом вызове из PI_NAME.
SFB21	RESUME	52 + длина указанных пользовательских данных при первом вызове из PI_NAME и ARG.
SFB22	STATUS	50
SFB23	USTATUS	50



## **20 Коммуникационные SFC для несконфигурированных S7-соединений**

## 20.1 Общие параметры коммуникационных SFC

### Входной параметр REQ

Входной параметр REQ (request to activate [запрос на активацию задания]) – это параметр управления, запускаемый уровнем сигнала. Он используется для запуска задания (по передаче данных или на прерывание соединения).

- При активации задания вызывается соответствующая функция SFC (с параметром REQ = 1), неактивная в текущий момент времени. Если при первом вызове SFC соединение с партнером по связи не существует, то прежде чем начнется передача данных, устанавливается соединение.
- Если Вы повторно запускаете задание, которое уже выполняется (и еще не завершено), то SFC не проверяет состояние параметра REQ.

### Входной параметр REQ\_ID (только для SFC 65 и SFC 66)

Входной параметр REQ\_ID используется для идентификации передаваемых данных. Он передается операционной системой передающего CPU для SFC 66 "X\_RCV" принимающего CPU партнера по связи.

Вы должны организовать запрос параметра REQ\_ID на принимающей стороне

- если Вы вызываете несколько SFC 65 "X\_SEND" с различными значениями параметра REQ\_ID на одном CPU передающей стороны и пересылаете данные партнеру по связи;
- если Вы используете SFC 65 "X\_SEND" для передачи данных одному партнеру по связи от нескольких CPU передающей стороны.

С помощью проверки REQ\_ID Вы можете сохранить принятые данные в различных областях памяти.

### Выходные параметры RET\_VAL и BUSY

Коммуникационные SFC выполняются асинхронно; это означает, что во время выполнения задания возможны несколько вызовов SFC. Выходные параметры RET\_VAL и BUSY показывают состояние задания. См. раздел "Значения параметров REQ, RET\_VAL и BUSY для асинхронных SFC".



## Входной параметр CONT

Входной параметр CONT (continue [продолжить]) – это параметр управления. Используя этот параметр, Вы выбираете, останется установленным или нет соединение с партнером по связи после того, как задание будет завершено.

- Если при первом вызове Вы выбираете CONT=0, то после завершения передачи данных соединение прерывается. Затем соединение снова доступно для обмена данными с новым партнером по связи.

Этот вариант гарантирует, что ресурсы соединения являются занятыми только на время фактического использования в конкретных заданиях.

- Если при первом вызове Вы выбираете CONT=1, то после завершения передачи данных соединение остается установленным.

Этот вариант полезен тогда, например, когда Вы выполняете циклический обмен данными между двумя станциями.

---

### Примечание

Соединение, установленное при CONT= 1, может быть явно прервано с помощью SFC69 "X\_ABORT" или SFC74 "I\_ABORT".

---

## 20.2 Информация об ошибках коммуникационных SFC для неконфигурированных S7-соединений

### Информация об ошибках

"Реальная" информация об ошибках для функций SFC 65...SFC 74 в соответствии с таблицей "Конкретная информация об ошибках для SFC 65...SFC 74" может быть классифицирована следующим образом:

Код ошибки (W#16#...)	Объяснение
809x	Ошибка в CPU, в котором выполняется SFC.
80Ax	Постоянная ошибка связи.
80Bx	Ошибка в партнере по связи.
80Cx	Нерегулярная ошибка.

Конкретная информация об ошибках для SFC 65 ... SFC 74.

Код ошибки (W#16# ...)	Объяснение (общее)	Объяснение (для конкретного SFC)
0000	Выполнение завершено без ошибок.	SFC69 "X_ABORT" и SFC74 "I_ABORT": REQ=1 и указанное соединение не установлено. SFC66 "X_RCV": EN_DT=1 и RD=NIL
00ху	-	SFC66 "X_RCV" при NDA=1 и RD<>NIL: RET_VAL содержит длину принятых данных (при EN_DT=0) или длину данных, скопированных в RD (при EN_DT=1). SFC67 "X_GET": RET_VAL содержит длину принятого блока данных. SFC72 "I_GET": RET_VAL содержит длину принятого блока данных.
7000	-	SFC65 "X_SEND", SFC67 "X_GET", SFC68 "X_PUT", SFC69 "X_ABORT", SFC72 "I_GET", SFC73 "I_PUT" и SFC74 "I_ABORT": Вызов с REQ = 0 (вызов без выполнения), BUSY имеет значение 0, передача данных не активна. SFC66 "X_RCV": EN_DT=0/1 и NDA=0
7001	Первый вызов с REQ=1: была запущена передача данных; BUSY имеет значение 1.	-
7002	Промежуточный вызов (REQ не имеет значения): передача данных уже активна; BUSY имеет значение 1.	SFC69 "X_ABORT" и SFC74 "I_ABORT": промежуточный вызов с REQ=1

Код ошибки (W#16# ...)	Объяснение (общее)	Объяснение (для конкретного SFC)
8090	Указанный адрес назначения партнера по связи недействителен, например: <ul style="list-style-type: none"> <li>• неправильный IOID</li> <li>• неправильный базовый адрес</li> <li>• неправильный адрес MPI (&gt; 126)</li> </ul>	-
8092	Ошибка в SD или RD, например: адресация локальной области данных не разрешена.	<p>SFC65 "X_SEND", например:</p> <ul style="list-style-type: none"> <li>• недопустимая длина для SD</li> <li>• SD=NIL недопустимо.</li> </ul> <p>SFC66 "X_RCV", например:</p> <ul style="list-style-type: none"> <li>• было принято большее количество данных, чем, может поместиться в области, заданной через RD</li> <li>• RD имеет тип данных BOOL, но принятые данные длиннее байта.</li> </ul> <p>SFC67 "X_GET" и SFC72 "I_GET", например:</p> <ul style="list-style-type: none"> <li>• недопустимая длина для RD</li> <li>• длина или тип данных RD не соответствует принятым данным</li> <li>• RD=NIL не разрешено.</li> </ul> <p>SFC68 "X_PUT" и SFC73 "I_PUT", например:</p> <ul style="list-style-type: none"> <li>• недопустимая длина для SD</li> <li>• SD=NIL недопустимо.</li> </ul>
8095	Блок уже выполняется в более низком классе приоритета.	-
80A0	Ошибка в принятом подтверждении.	SFC68 "X_PUT" и SFC73 "I_PUT": тип данных, указанный в SD передающего CPU, не поддерживается партнером по связи.
80A1	Проблемы связи: вызов SFC после прерывания существующего соединения.	-
80B0	Объект недоступен, например, незагруженный DB.	Возможно в случае SFC67 "X_GET" и SFC68 "X_PUT" и SFC72 "I_GET" и SFC73 "I_PUT".
80B1	Ошибка в указателе ANY. Неправильная длина области передаваемых данных.	-
80B2	Аппаратная ошибка: модуль не существует. <ul style="list-style-type: none"> <li>• Конфигурированный слот не занят.</li> <li>• Фактический тип модуля не соответствует ожидаемому типу.</li> <li>• Децентрализованная периферия недоступна.</li> <li>• Нет записи для модуля в соответствующем SDB.</li> </ul>	Возможно в случае SFC67 "X_GET" и SFC68 "X_PUT" и SFC72 "I_GET" и SFC73 "I_PUT".
80B3	Данные могут или только читаться, или только записываться, например, DB, защищенный от записи.	Возможно в случае SFC67 "X_GET" и SFC68 "X_PUT" и SFC72 "I_GET" и SFC73 "I_PUT".

Код ошибки (W#16# ...)	Объяснение (общее)	Объяснение (для конкретного SFC)
80B4	Ошибка типа данных в указателе ANY, или массив заданного типа не разрешен.	SFC67 "X_GET" и SFC68 "X_PUT" и SFC72 "I_GET" и SFC73 "I_PUT": тип данных, заданный в VAR_ADDR, не поддерживается партнером по связи.
80B5	Выполнение отклонено из-за недопустимого режима.	Возможно в случае SFC65 "X_SEND".
80B6	Принятое подтверждение содержит неизвестный код ошибки.	-
80B7	Тип данных и/или длина переданных данных не соответствуют области в CPU партнера, в которой они должны записываться.	Возможно в случае SFC68 "X_PUT" и SFC73 "I_PUT".
80B8	-	SFC65 "X_SEND": SFC66 "X_RCV" партнера по связи не позволил принять данные (RD=NIL).
80B9	-	SFC65 "X_SEND": Блок данных был идентифицирован партнером по связи (вызов SFC66 "X_RCV" с EN_DT=0), он еще не был введен в программу пользователя, потому что партнер находится в режиме STOP.
80BA	Ответ партнера по связи не помещается в кадре связи.	-
80C0	Указанное соединение используется другим заданием.	-
80C1	Недостаток ресурсов в CPU, в котором выполняется SFC, например: В модуле уже выполняется максимальное количество разных заданий на передачу. Ресурс соединения использован, например, для приема данных.	-
80C2	Временный недостаток ресурсов в партнере по связи, например: <ul style="list-style-type: none"> <li>Партнер по связи в настоящее время обрабатывает максимальное число заданий.</li> <li>Требуемые ресурсы, память, и т.д. используются.</li> <li>Недостаточно рабочей памяти (сжать память).</li> </ul>	-
80C3	Ошибка в установлении соединения, например: <ul style="list-style-type: none"> <li>Локальная станция S7 не подключена к подсети MPI.</li> <li>Вы обратились в подсети MPI к своей собственной станции.</li> <li>Партнер по связи больше не доступен.</li> <li>Временный недостаток ресурсов в партнере по связи.</li> </ul>	-

## 20.3 Передача данных партнеру по связи, находящемуся вне локальной станции S7, с помощью SFC65 "X\_SEND"

### Описание

С помощью SFC65 "X\_SEND" Вы передаете данные партнеру по связи вне локальной станции S7. Данные принимаются партнером по связи с помощью SFC66 "X\_RCV".

Вы можете идентифицировать Ваши данные с помощью входного параметра REQ\_ID. Этот идентификатор (ID) задания передается с данными. Вы можете оценивать этот параметр в партнере по связи, чтобы определить источник данных.

Данные передаются после вызова SFC с REQ=1.

Убедитесь, что область передачи, определенная параметром SD (в передающем CPU) имеет меньший или точно такой же размер, что и область приема, определенная параметром RD (в партнере по связи). Если SD имеет тип данных BOOL, то RD тоже должен иметь тип данных BOOL.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры SFB/FB и SFC/FC для базовой системы связи S7".
CONT	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "продолжить". См. "Общие параметры SFB/FB и SFC/FC для базовой системы связи S7".
DEST_ID	INPUT	WORD	I, Q, M, D, L, константа	Адресный параметр "ID адресата". Он содержит MPI-адрес партнера по связи. Конфигурируется с помощью STEP 7.
REQ_ID	INPUT	DWORD	I, Q, M, D, L, константа	Идентификатор задания. Он используется для идентификации данных в партнере по связи.
SD	INPUT	ANY	I, Q, M, D	Ссылка на область для передачи. Разрешены следующие типы данных: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME и массивы этих типов данных, кроме BOOL.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Передача не завершена. BUSY=0: Передача завершена или функция передачи не активна.

### **Входной параметр REQ\_ID**

Входной параметр REQ\_ID используется для идентификации Ваших данных. Он передается операционной системой передающего CPU функции SFC66 "X\_RCV" в CPU партнера по связи.

Параметр REQ\_ID нужен на приемном конце в следующих ситуациях:

- Когда Вы вызываете более одного блока SFC65 "X\_SEND" с различными параметрами REQ\_ID в передающем CPU и передаете данные одному партнеру по связи.
- Когда Вы передаете данные одному партнеру по связи из более, чем одного передающего CPU, используя SFC65 "X\_SEND".

Оценивая REQ\_ID, Вы можете сохранять принимаемые данные в различных областях памяти.

### **Консистентность данных**

Данные передаются в консистентном состоянии.

### **Информация об ошибках**

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.4 Прием данных от партнера по связи, находящегося вне локальной станции S7, с помощью SFC66 "X\_RCV"

### Описание

С помощью SFC66 "X\_RCV" можно принимать данные, передаваемые одним или несколькими партнерами по связи с помощью SFC65 "X\_SEND", находящимися вне локальной станции S7.

С помощью SFC66 "X\_RCV"

- Вы можете проверить, переданы ли данные и ожидают ли копирования. Данные вводятся во внутреннюю очередь операционной системой.
- Вы можете копировать самый "старый" блок данных из очереди в выбранную область памяти для приема.

Параметр	Описание	Тип данных	Область данных	Характеристика
EN_DT	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разблокировать передачу данных". С помощью значения 0 Вы можете проверить, ожидает ли, по крайней мере, один блок данных ввода в область приема. При EN_DT = 1 копируется самый "старый" блок данных из очереди в область рабочей памяти, заданную в RD.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время выполнения функции происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки. Если ошибки не происходит, то RET_VAL содержит следующее: <ul style="list-style-type: none"> <li>• W#16#7000, если EN_DT=0/1 и NDA=0. В этом случае в очереди нет блоков данных.</li> <li>• длину самого "старого" блока данных, введенного в очередь, в форме положительного числа в байтах, если EN_DT=0 и NDA=1.</li> <li>• длину блока данных, скопированного в область приема RD, в форме положительного числа в байтах, если EN_DT=1 и NDA=1.</li> </ul>
REQ_ID	OUTPUT	DWORD	I, Q, M, D, L	Идентификатор задания SFC "X_SEND", чьи данные стоят первыми в очереди, иными словами, самые "старые" данные в очереди. Если в очереди нет блоков данных, то REQ_ID имеет значение 0.

Параметр	Описание	Тип данных	Область данных	Характеристика
NDA	OUTPUT	BOOL	I, Q, M, D, L	<p>Параметр состояния "поступили новые данные".</p> <p>NDA=0:</p> <ul style="list-style-type: none"> <li>В очереди нет блоков данных.</li> </ul> <p>NDA=1:</p> <ul style="list-style-type: none"> <li>Очередь содержит, по крайней мере, один блок данных. (Вызов SFC66 с EN_DT=0).</li> <li>Самый "старый" блок данных в очереди был скопирован в программу пользователя. (Вызов SFC66 с EN_DT=1).</li> </ul>
RD	OUTPUT	ANY	I, Q, M, D	<p>Ссылка на область принимаемых данных. Разрешены следующие типы данных: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME и массивы этих типов данных, кроме BOOL.</p> <p>Если Вы хотите отбросить самый "старый" блок данных в очереди, то присвойте RD значение NIL.</p>

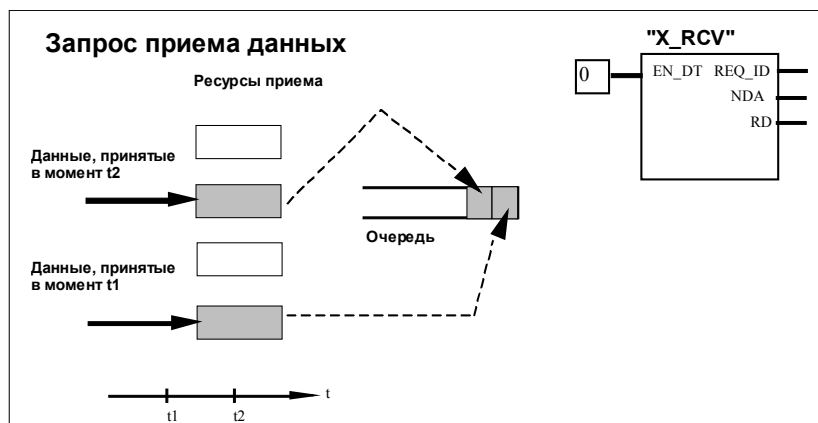
#### Индикация приема данных с помощью EN\_DT=0

Как только данные поступают от партнера по связи, они вводятся операционной системой в очередь в том порядке, в котором они приняты. Если Вы хотите проверить, находится ли в очереди хотя бы один блок данных, то вызовите SFC66 с EN\_DT=0 и оцените выходной параметр NDA следующим образом:

- NDA=0 означает, что очередь не содержит блоков данных. Параметр REQ\_ID является несущественным, RET\_VAL имеет значение W#16#7000.
- NDA=1 означает, что в очереди имеется, по крайней мере, один блок данных, который может быть извлечен.

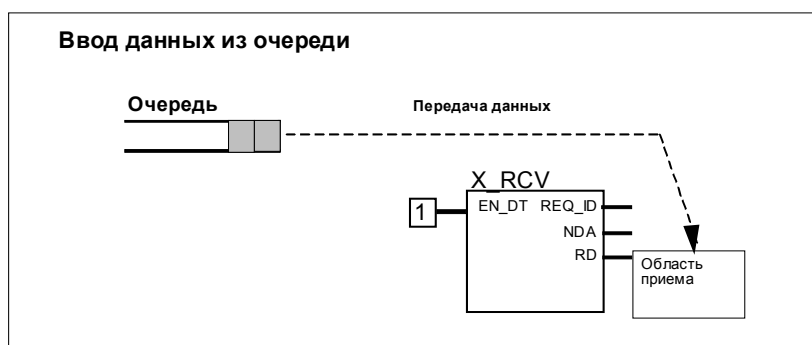
В этом случае Вы должны также оценить выходной параметр RET\_VAL и, если возможно, REQ\_ID. RET\_VAL содержит длину блока данных в байтах, REQ\_ID содержит идентификатор задания передающего блока. Если в очереди имеются несколько блоков данных, то REQ\_ID и RET\_VAL принадлежат самому "старому" блоку данных в очереди.





### Ввод данных в область приема с помощью EN\_DT=1

Когда Вы вызываете SFC66 "X\_RCV" с EN\_DT=1, самый старый блок данных из очереди копируется в область рабочей памяти, заданную посредством RD. RD должен быть по размеру больше или равен области передачи соответствующего SFC65 "X\_SEND", определенной параметром SD. Если входной параметр SD имеет тип данных BOOL, то RD также должен иметь тип данных BOOL. Если Вы хотите вводить принимаемые данные в разные области, то Вы можете запрашивать REQ\_ID (вызов SFC с EN\_DT = 0) и выбирать подходящий RD в продолженном вызове (при EN\_DT=1). Если во время копирования данных ошибки не происходит, то RET\_VAL содержит длину скопированного блока данных в байтах, и отправителю передается положительное подтверждение.



### Отказ от данных

Если Вы не хотите вводить данные из очереди, то присвойте RD значение NIL (см. [1232/](#)). В этом случае отправитель принимает отрицательное подтверждение (RET\_VAL в соответствующем SFC65 "X\_SEND" имеет значение W#1680B8). RET\_VAL в SFC66 "X\_RCV" имеет значение 0.

### Консистентность данных

Вы не должны считывать область приема, пока задание не завершено. Иначе Вы могли бы считать данные, которые не связаны друг с другом (противоречивые данные).

### Переключение в режим STOP

Если CPU переключается в режим STOP, то

- все вновь поступающие задания получают отрицательное подтверждение
  - все задания, которые поступили и находятся в очереди, получают отрицательное подтверждение.
    - Если после STOP следует теплый или холодный рестарт, то все блоки данных отбрасываются.
    - Если после STOP следует горячий рестарт (невозможный в S7-300 и S7-400H), то блок данных, принадлежащий самому старому заданию, вводится в программу пользователя при условии, что перед переключением в режим STOP очередь запрашивалась (посредством вызова SFC66 "X\_RCV" с EN\_DT=0). В противном случае он отбрасывается.
- Все другие блоки данных отбрасываются.

### Прерывание соединения

Если соединение прерывается, то принадлежащее соединению задание, которое уже находится в очереди, отбрасывается.

Исключение: Если это задание самое старое в очереди, и Вы уже обнаружили его присутствие, вызвав SFC66 "X\_RCV" с EN\_DT=0, то Вы можете ввести его в область приема с помощью EN\_DT=1.

### Информация об ошибках

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.5 Запись данных в партнере по связи вне локальной станции S7 с помощью SFC68 "X\_PUT"

### Описание

С помощью SFC68 "X\_PUT" Вы записываете данные в партнере по связи, который не находится в той же самой локальной станции S7. В партнере по связи нет соответствующего SFC.

Задание на запись активируется после вызова SFC с REQ=1. После этого Вы продолжаете вызывать SFC до тех пор, пока не будет принято подтверждение в виде BUSY=0.

Убедитесь, что определенная параметром SD область передачи (в передающем CPU) имеет такую же длину, как определенная параметром VAR\_ADDR область приема (в партнере по связи). Типы данных SD и VAR\_ADDR также должны совпадать.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры для SFC для базовой системы связи S7".
CONT	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "продолжить". См. "Общие параметры для SFC для базовой системы связи S7".
DEST_ID	INPUT	WORD	I, Q, M, D, L, константа	Адресный параметр "ID адресата". Он содержит MPI-адрес партнера по связи. Вы сконфигурировали его с помощью STEP 7.
VAR_ADDR	INPUT	ANY	I, Q, M, D	Ссылка на область в CPU партнера, в которую будут записываться данные. Вы должны выбрать тип данных, который поддерживается партнером по связи.
SD	INPUT	ANY	I, Q, M, D	Ссылка на область локального CPU, содержащую передаваемые данные. Разрешены следующие типы данных: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME и массивы этих типов данных, кроме BOOL. SD должен иметь такую же длину, как параметр VAR_ADDR партнера по связи. Типы данных SD и VAR_ADDR также должны совпадать.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Передача еще не завершена. BUSY=0: Передача завершена или функция передачи не активна.

### **Переключение в режим STOP**

Если CPU переключается в режим STOP, то установленное SFC68 "X\_PUT" соединение прерывается. Данные больше не могут передаваться. Если переданные данные уже были скопированы во внутренний буфер до смены режима CPU, то содержимое буфера отбрасывается.

### **Партнер переключается в режим STOP**

Если CPU партнера по связи переключается в режим STOP, то это не влияет на передачу данных с помощью SFC68 "X\_PUT". Данные могут записываться также и тогда, когда партнер находится в состоянии STOP.

### **Консистентность данных**

Данные пересылаются в консистентном состоянии.

### **Информация об ошибках**

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.6 Чтение данных из партнера по связи, находящегося вне локальной станции S7, с помощью SFC67 "X\_GET"

### Описание

С помощью SFC67 "X\_GET" Вы можете считать данные партнера по связи, который находится вне локальной станции S7. В партнере по связи нет соответствующего SFC.

Задание на чтение активируется после вызова SFC с REQ=1. После этого Вы продолжаете вызывать SFC до тех пор, пока посредством BUSY=0 не отобразится прием данных. Тогда RET\_VAL содержит длину принятого блока данных в байтах.

Убедитесь, что определенная параметром RD область приема (в приемном CPU) имеет, по крайней мере, такую же длину, как подлежащая чтению область (в партнере по связи), определенная параметром VAR\_ADDR. Типы данных RD и VAR\_ADDR также должны совпадать.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры для SFC для базовой системы связи S7".
CONT	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "продолжать". См. "Общие параметры для SFC для базовой системы связи S7".
DEST_ID	INPUT	WORD	I, Q, M, D, L, константа	Адресный параметр "ID адресата". Он содержит MPI-адрес партнера по связи. Вы сконфигурировали его с помощью STEP 7.
VAR_ADDR	INPUT	ANY	I, Q, M, D	Ссылка на область в CPU партнера, из которой будут читаться данные. Вы должны выбрать тип данных, который поддерживается партнером по связи.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки. Если ошибки не происходит, то RET_VAL содержит длину блока данных, скопированного в область приема RD, в форме положительного числа байтов.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Прием еще не завершен. BUSY=0: Прием завершен или нет активного задания на прием.

Параметр	Описание	Тип данных	Область данных	Характеристика
RD	OUTPUT	ANY	I, Q, M, D	Ссылка на область приема (область принимаемых данных). Разрешены следующие типы данных: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME и массивы этих типов данных, кроме BOOL. Область приема RD должна иметь, по крайней мере, такую же длину, как определенная параметром VAR_ADDR область, из которой читаются данные. Типы данных RD и VAR_ADDR также должны совпадать

### Переключение в режим STOP

Если CPU переключается в режим STOP, то установленное SFC67 "X\_GET" соединение прерывается. От типа выполняемого рестарта зависит, будут ли потеряны принятые данные, расположенные в буфере операционной системы:

- После горячего рестарта (нет в S7-300 и S7-400H) данные копируются в область, определяемую RD.
- После теплого или холодного рестарта данные отбрасываются.

### Партнер переключается в режим STOP

Если CPU партнера по связи переключается в режим STOP, то это не влияет на передачу данных с помощью SFC67 "X\_GET". Данные могут читаться также и тогда, когда партнер находится в состоянии STOP.

### Консистентность данных

Данные пересылаются в консистентном состоянии.

### Информация об ошибках

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.7 Прерывание существующего соединения с партнером по связи, находящимся вне локальной станции S7, с помощью SFC69 "X\_ABORT"

### Описание

С помощью SFC69 "X\_ABORT" Вы прерываете соединение, установленное SFC X\_SEND, X\_GET или X\_PUT с партнером по связи, который не находится в той же самой локальной станции S7. Если задание, принадлежащее X\_SEND, X\_GET или X\_PUT, завершено (BUSY = 0), то после вызова SFC69 "X\_ABORT" ресурсы соединения, используемые на обоих концах, освобождаются. Если задание, принадлежащее X\_SEND, X\_GET или X\_PUT, еще не завершено (BUSY = 1), то после того, как соединение будет прервано, вновь вызовите соответствующий SFC с REQ = 0 и CONT = 0 и затем ожидайте BUSY = 0. Только тогда все ресурсы соединения вновь освобождаются. Вы можете вызывать SFC69 "X\_ABORT" только на том конце, где расположены SFC "X\_SEND", "X\_PUT" или "X\_GET". Прерывание соединения активируется посредством вызова SFC с REQ=1.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры для SFC для базовой системы связи S7".
DEST_ID	INPUT	WORD	I, Q, M, D, L, константа	Адресный параметр "ID адресата". Он содержит MPI-адрес партнера по связи. Вы сконфигурировали его с помощью STEP 7.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Прерывание соединения еще не завершено. BUSY=0: Прерывание соединения завершено.

### Переключение в режим STOP

Если CPU переключается в режим STOP, то прерывание соединения, запущенное посредством SFC69 "X\_ABORT", завершается.

### Партнер переключается в режим STOP

Если CPU партнера по связи переключается в режим STOP, то это не влияет на прерывание соединения с помощью SFC69 "X\_ABORT". Соединение прерывается.

### Информация об ошибках

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.8 Чтение данных из партнера по связи, находящегося в пределах локальной станции S7, с помощью SFC72 "I\_GET"

### Описание

С помощью SFC72 "I\_GET" Вы можете читать данные из партнера по связи в той же самой локальной станции S7. Партнер по связи может находиться в центральной стойке, в стойке расширения или быть децентрализованным. Убедитесь, что Вы назначили децентрализованных коммуникационных партнеров локальному CPU с помощью STEP 7. В партнере по связи нет соответствующего SFC.

Задание на прием активируется после вызова SFC с REQ=1. После этого Вы продолжаете вызывать SFC до тех пор, пока посредством BUSY=0 не отобразится прием данных. Тогда RET\_VAL содержит длину принятого блока данных в байтах.

Убедитесь, что определенная параметром RD область приема (в принимающем CPU) имеет, по крайней мере, такую же длину, как подлежащая чтению область (в партнере по связи), определенная параметром VAR\_ADDR. Типы данных RD и VAR\_ADDR также должны совпадать.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры для SFC для базовой системы связи S7".
CONT	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "продолжать". См. "Общие параметры для SFC для базовой системы связи S7".
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресного диапазона в модуле партнера: V#16#54= Периферийный вход (PI) V#16#55= Периферийный выход (PQ) Идентификатором диапазона, принадлежащего смешанному модулю, является меньший из двух адресов. Если адреса одинаковые, то задайте V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес модуля партнера. Если это смешанный модуль, то задайте меньший из двух адресов.
VAR_ADDR	INPUT	ANY	I, Q, M, D	Ссылка на область в CPU партнера, из которой будут читаться данные. Выберите тип данных, поддерживаемый партнером по связи.



Параметр	Описание	Тип данных	Область данных	Характеристика
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки. Если ошибки не происходит, то RET_VAL содержит длину блока данных, скопированного в область приема RD, в форме положительного числа байтов.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Прием еще не завершен. BUSY=0: Прием завершен или нет активного задания на прием.
RD	OUTPUT	ANY	I, Q, M, D	Ссылка на область приема (область принимаемых данных). Разрешены следующие типы данных: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME и массивы этих типов данных, кроме BOOL. Область приема RD должна иметь, по крайней мере, такую же длину, как определенная параметром VAR_ADDR область, из которой читаются данные. Типы данных RD и VAR_ADDR также должны совпадать.

### Переключение в режим STOP

Если CPU переключается в режим STOP, то установленное SFC72 "I\_GET" соединение прерывается. От типа выполняемого рестарта зависит, будут ли потеряны принятые данные, расположенные в буфере операционной системы:

- После горячего рестарта (нет в S7-300 и S7-400H) данные копируются в область, определяемую RD.
- После теплого или холодного рестарта данные стираются.

### Переключение партнера по связи в режим STOP

Если CPU партнера по связи переключается в режим STOP, то это не влияет на передачу данных с помощью SFC72 "I\_GET". Данные могут читаться также и тогда, когда партнер находится в состоянии STOP.

### Консистентность данных

Данные пересылаются в консистентном состоянии.

### Информация об ошибках

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.9 Запись данных в партнере по связи, находящемся в пределах локальной станции S7, с помощью SFC73 "I\_PUT"

### Описание

С помощью SFC73 "I\_PUT" Вы записываете данные в партнере по связи, который находится в той же самой локальной станции S7. Партнер по связи может быть в центральной стойке, в стойке расширения или быть децентрализованным. Убедитесь, что Вы назначили децентрализованных коммуникационных партнеров локальному CPU с помощью STEP 7. В партнере по связи нет соответствующего SFC.

Задание на передачу активируется после вызова SFC при сигнальном уровне 1 на управляющем входе REQ.

Убедитесь, что определенная параметром SD область передачи (в передающем CPU) имеет такую же длину, как определенная параметром VAR\_ADDR область приема (в партнере по связи). Типы данных SD и VAR\_ADDR также должны совпадать.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры для SFC для базовой системы связи S7".
CONT	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "продолжать". См. "Общие параметры для SFC для базовой системы связи S7".
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресного диапазона в модуле партнера: V#16#54= Периферийный вход (PI) V#16#55= Периферийный выход (PQ) Идентификатором диапазона, принадлежащего смешанному модулю, является меньший из двух адресов. Если адреса одинаковые, то задайте V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес модуля партнера. Если это смешанный модуль, то задайте меньший из двух адресов.
VAR_ADDR	INPUT	ANY	I, Q, M, D, L	Ссылка на область в партнере по связи, в которую будут записываться данные. Выберите тип данных, который поддерживается партнером по связи.

Параметр	Описание	Тип данных	Область данных	Характеристика
SD	INPUT	ANY	I, Q, M, D	Ссылка на область локального CPU, содержащую передаваемые данные. Разрешены следующие типы данных: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME и массивы этих типов данных, кроме BOOL. SD должен иметь такую же длину, как параметр VAR_ADDR партнера по связи. Типы данных SD и VAR_ADDR также должны совпадать.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Передача еще не завершена. BUSY=0: Передача завершена или функция передачи не активна.

### Переключение в режим STOP

Если CPU переключается в режим STOP, то установленное SFC73 "I\_PUT" соединение прерывается. Данные больше не могут передаваться. Если переданные данные уже были скопированы во внутренний буфер до смены режима CPU, то содержимое буфера отбрасывается.

### Партнер переключается в режим STOP

Если CPU партнера по связи переключается в режим STOP, то это не влияет на передачу данных с помощью SFC73 "I\_PUT". Данные могут записываться также и тогда, когда партнер находится в состоянии STOP.

### Консистентность данных

Данные пересылаются в консистентном состоянии.

### Информация об ошибках

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".

## 20.10 Прерывание существующего соединения с партнером по связи, находящимся в пределах локальной станции S7, с помощью SFC74 "I\_ABORT"

### Описание

С помощью SFC74 "I\_ABORT" Вы прерываете соединение, установленное SFC72 "I\_GET" или SFC73 "I\_PUT" с партнером по связи в той же самой локальной станции S7. Если задание, принадлежащее I\_GET или I\_PUT, завершено (BUSY = 0), то после вызова SFC74 "I\_ABORT" ресурсы соединения, используемые на обоих концах, освобождаются.

Если задание, принадлежащее I\_GET или I\_PUT, еще не завершено (BUSY = 1), то после того, как соединение будет прервано, вызовите вновь соответствующий SFC с REQ = 0 и CONT = 0 и затем ожидайте BUSY = 0. Только тогда все ресурсы соединения вновь освобождаются.

Вы можете вызывать SFC74 "I\_ABORT" только на том конце, где расположен SFC "I\_PUT" или "I\_GET" (иными словами на стороне клиента).

Прерывание соединения активируется посредством вызова SFC с REQ=1.

Параметр	Описание	Тип данных	Область данных	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "запрос на выполнение задания". См. "Общие параметры для SFC для базовой системы связи S7".
IOID	INPUT	BYTE	I, Q, M, D, L, константа	Идентификатор адресного диапазона в модуле партнера: V#16#54= Периферийный вход (PI) V#16#55= Периферийный выход (PQ) Идентификатором диапазона, принадлежащего смешанному модулю, является меньший из двух адресов. Если адреса одинаковые, то задайте V#16#54.
LADDR	INPUT	WORD	I, Q, M, D, L, константа	Логический адрес модуля партнера. Если это смешанный модуль, то задайте меньший из двух адресов.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если в то время, когда функция выполняется, происходит ошибка, то возвращаемое значение содержит соответствующий код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Прерывание соединения еще не завершено. BUSY=0: Прерывание соединения завершено.

### **Переключение в режим STOP**

Если CPU переключается в режим STOP, то прерывание соединения, запущенное посредством SFC74 "I\_ABORT", завершается.

### **Партнер переключается в режим STOP**

Если CPU партнера по связи переключается в режим STOP, то это не влияет на прерывание соединения с помощью SFC74 "I\_ABORT". Соединение прерывается.

### **Информация об ошибках**

См. раздел "Информация об ошибках для коммуникационных SFC для неконфигурированных S7-соединений".



## 21 PROFINet

## 21.1 Основные сведения, касающиеся SFC 112, 113 и 114

### Примечание

По умолчанию интерфейс PROFInet обновляет операционная система. Выполняет это она так же, как соединения в системе распределенной периферии (DP interconnections) - в определенные моменты цикла сканирования. Однако, если Вы отключили автоматическое обновление в Вашей конфигурации (например, чтобы уменьшить нагрузку на CPU), То Вы должны будете выполнять обновления самостоятельно. Для этого Вы должны вызывать системные функции SFC 112 ... SFC 114 в определенные моменты времени.

Обновления, упоминаемые здесь, могут быть отключены только вместе в группе в конфигурации.

### "Теневая" память (Shadow Memory)

Интерфейсный DB (interface DB) - это интерфейс пользовательской программы для компонентов PROFInet. Для того, чтобы обеспечить консистентность входов (inputs) и выходов (output) при выполнении соответствующих разделов программы, для каждого интерфейсного DB (interface DB) имеется идентично структурированная область памяти, которая управляется операционной системой. Эта область называется "теневой" памятью ("shadow memory") В вашей программе Вы имеете доступ к интерфейсному DB (interface DB), а другие (внешние) компоненты PROFInet имеют доступ только к "теневой" памяти. Такая организация предотвращает конфликты во время операций доступа ко входам (inputs) и выходам (outputs) интерфейсного DB (interface DB).

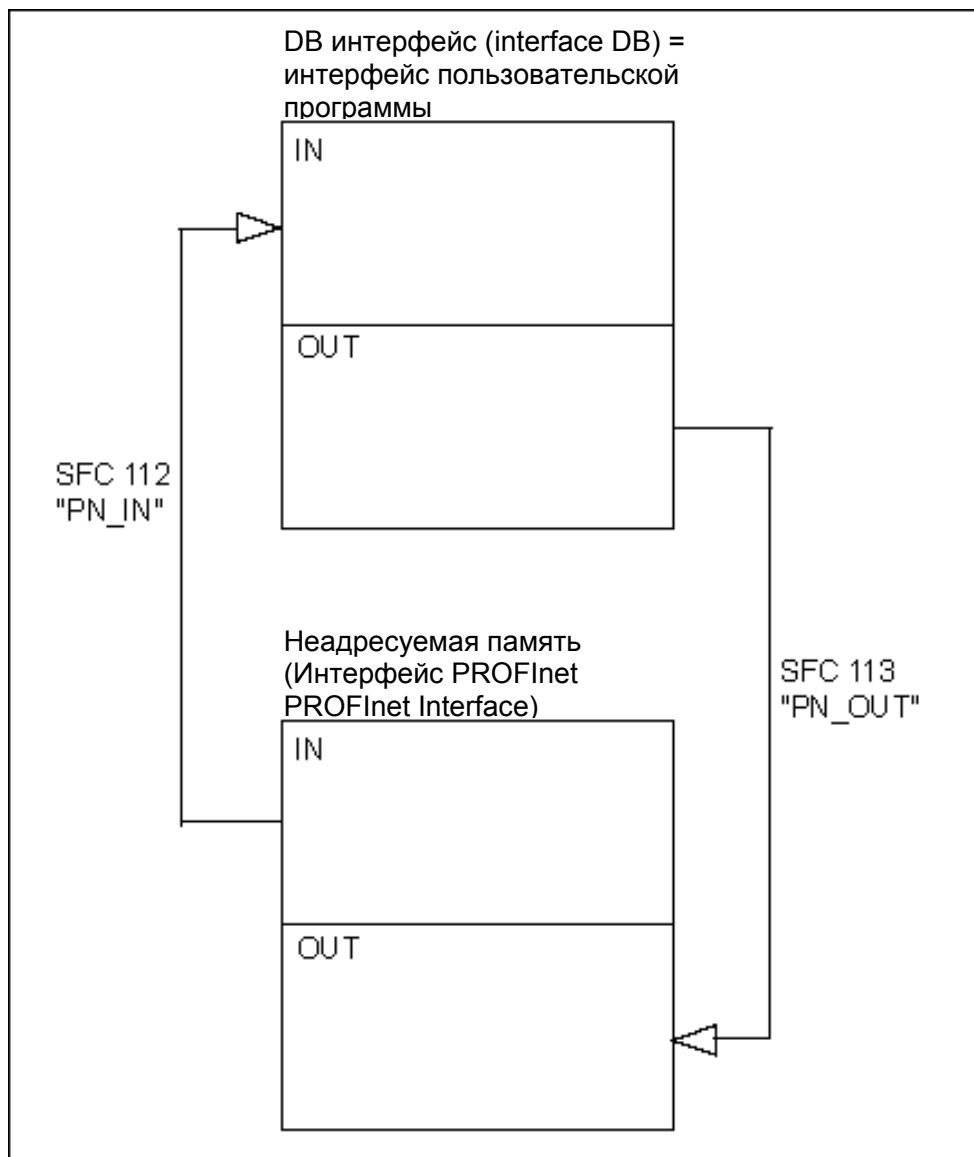
### Обновление интерфейса PROFInet

Консистентность данных обеспечивается с помощью использования "теневой" памяти ("shadow" memory), что означает, что обновление интерфейса PROFInet происходит в течение следующих двух этапов:

- Входы (inputs) "теневой" памяти ("shadow" memory) копируются во входы (inputs) интерфейсного DB перед тем, как будет запущена Ваша программа для компонентов PROFInet.
- Выходы (outputs) интерфейсного DB копируются в выходы (outputs) "теневой" памяти ("shadow" memory) после того, как Ваша программа для компонентов PROFInet была обработана.

Интерфейс PROFInet обновляется или операционной системой, или с помощью SFC 112 и SFC 113, в зависимости от того, как Вы сконфигурировали компоненты PROFInet в SIMATIC Manager. Следующие иллюстрации показывают схему выполнения обновления с помощью SFC 112 и 113.

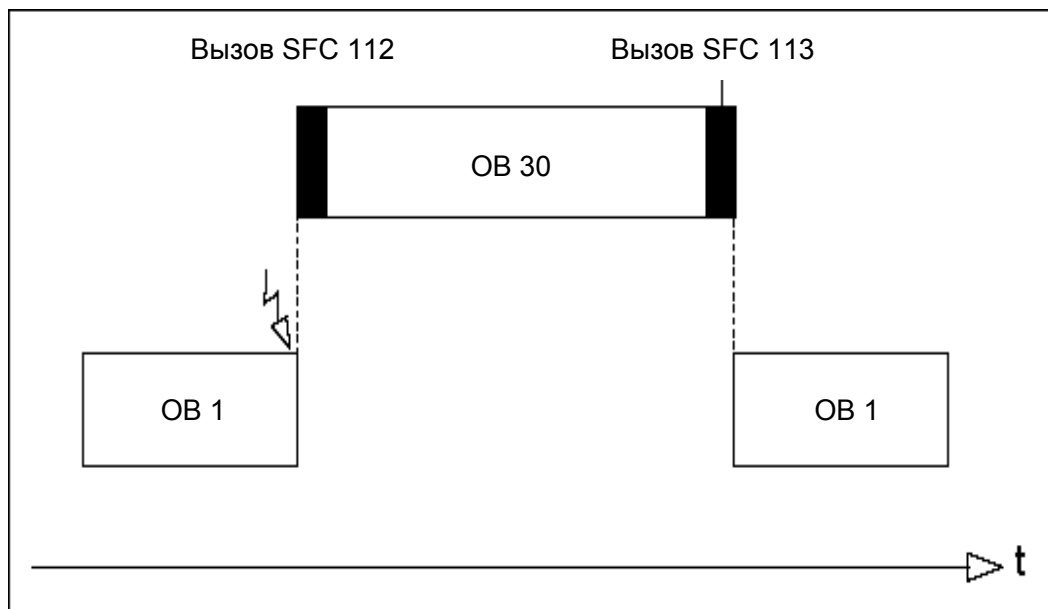




Если операционная система обновляет интерфейс PROFInet, то она выполняет его в определенные моменты цикла сканирования.

Если обновление выполняет пользователь с помощью системных функций SFC 112 и SFC 113, то Вы должны будете вызывать SFC 112 при запуске OB, содержащего программу, связанную с компонентами PROFInet, и вызывать SFC113 в конце обработки этого блока OB.

Примером такой процедуры может быть пример с использованием OB30, представленный на следующем рисунке.



### Обновление соединений в системе распределенной периферии (DP interconnections)

Обновление соединений в системе распределенной периферии (DP interconnections) производится или операционной системой, или пользователем с помощью системной функции SFC114, в зависимости от того, как Вы сконфигурировали компоненты PROFInet в SIMATIC Manager.

## 21.2 Обновление входов UPI для компонентов PROFInet с помощью SFC112 "PN\_IN"

### Описание

Используя системную функцию SFC112 "PN\_IN", Вы можете скопировать данные входов (input) в PROFInet из "теневого" (shadow) памяти для компонентов PROFInet в связанный интерфейсный DB (interface DB). После выполнения SFC Ваша программа будет иметь доступ к текущим значениям для входов (input).

Параметр	Описание	Тип данных	Область памяти	Характеристика
DBNO	INPUT	WORD	I, Q, M, D, L, constant	DB не является интерфейсным DB
RET_VAL	OUTPUT	INT	I, Q, M, D, L,	Информация об ошибках

### Информация по ошибкам

Код ошибки (W#16#...)	Пояснение
0000	Нет ошибок.
8002	Номер DB не соответствует ни одному из компонентов конфигурации.
8004	Номер DB не соответствует ни одному из компонентов конфигурации, но этот DB не загружен.
8006	Интерфейсный DB защищен от записи (write-protected) в CPU.
80B1	Ошибка размера записи при чтении или записи. Конфигурация компонента не записывается в загруженный DB.
8хху	Общая информация об ошибках, см. <i>Проверка ошибок с выходным параметром RET_VAL</i>

### Примечание

Обратитесь также к разделу *Основные сведения, касающиеся SFC 112, 113 и 114*

## 21.3 Обновление выходов UPI для компонентов PROFInet с помощью SFC113 "PN\_OUT"

### Описание

Используя системную функцию SFC113 "PN\_OUT", Вы можете скопировать данные выходов (output), сгенерированные в Вашей программе, из интерфейсного блока DB (interface DB) для PROFInet компонентов в связанную "теневую" память (shadow memory). После обработки SFC компоненты PROFInet будут иметь доступ к текущим значениям для выходов (output).

Параметр	Описание	Тип данных	Область памяти	Характеристика
DBNO	INPUT	WORD	I, Q, M, D, L, constant	DB не является интерфейсным DB
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках

### Информация по ошибкам

Код ошибки (W#16#...)	Пояснение
0000	Нет ошибок.
8002	Номер DB не соответствует ни одному из компонентов конфигурации.
8004	Номер DB не соответствует ни одному из компонентов конфигурации, но этот DB не загружен.
8006	Интерфейсный DB защищен от записи (write-protected) в CPU.
80B1	Ошибка размера записи при чтении или записи. Конфигурация компонента не записывается в загруженный DB.
8хху	Общая информация об ошибках, см. <i>Проверка ошибок с выходным параметром RET_VAL</i>

### Примечание

Обратитесь также к разделу *Основные сведения, касающиеся SFC 112, 113 и 114*

## 21.4 Обновление соединений в системе распределенной периферии (DP interconnections) с помощью SFC114 "PN\_DP"

### Описание

Используя системную функцию SFC114 "PN\_DP" Вы можете обновлять данные для следующих соединений:

- Взаимные соединения (Interconnections) между компонентами PROFInet в локальной сети PROFIBUS
- Взаимные соединения (Interconnections) с циклической передачей данных между компонентами PROFInet в локальной сети PROFIBUS и внешними компонентами PROFInet. Взаимные соединения (Interconnections) между подсетями (между Industrial Ethernet и PROFIBUS DP).

### Функция

SFC114 "PN\_DP" функционирует в асинхронном режиме. Это означает, что ее выполнение распределяется на несколько вызовов SFC. Чтобы начать обновление взаимных соединений (DP interconnections), вызовите SFC114 с параметром REQ=1.

Состояние (статус) задания отображается в выходных параметрах RET\_VAL и BUSY (см. также *Значения параметров REQ, RET\_VAL и BUSY для асинхронного режима обработки SFC*).

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L, константа	REQ=1: начало обновления взаимных соединений в системе распределенной периферии DP (Interconnections DP)
RET_VAL	OUTPUT	INT	I, Q, M, D, L	При возникновении ошибки во время выполнения функции возвращаемое значение содержит соответствующий код ошибки.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Процесс обновления взаимных соединений в системе распределенной периферии DP (Interconnections DP) еще не завершен.

## Информация по ошибкам

Код ошибки (W#16#...)	Пояснение
0000	Нет ошибок.
7000	REQ = 0 при первом вызове: Процесс обновления взаимных соединений в системе распределенной периферии DP (Interconnections DP) не начат. BUSY имеет значение 0.
7001	REQ = 1 при первом вызове: BUSY имеет значение 1.
7002	Последующий вызов (параметр REQ некорректен). Процесс обновления взаимных соединений в системе распределенной периферии DP (Interconnections DP) еще не завершен. Параметр BUSY имеет значение 1.
8095	Вы инициировали другой процесс обновления взаимных соединений в системе распределенной периферии DP в более высоком приоритетном классе. Тем не менее, процесс обновления взаимных соединений в системе распределенной периферии DP в более низком приоритетном классе (выполняемый операционной системой или с помощью функции SFC114) продолжается.
8хху	Общая информация об ошибках, см. проверка ошибок с выходным параметром RET_VAL

## Примечание

Обратитесь также к разделу *Основные сведения, касающиеся SFC 112, 113 и 114*

## **22 Создание сообщений, связанных с блоками**

## 22.1 Введение в создание сообщений, связанных с блоками, с помощью SFB

### SFB для создания сообщений, связанных с блоками

Вы можете создать сообщение, связанное с блоком, вызывая в своей программе один из следующих SFB:

- SFB36 "NOTIFY"
- SFB31 "NOTIFY\_8P"
- SFB33 "ALARM"
- SFB35 "ALARM\_8P"
- SFB34 "ALARM\_8"

Эти SFB имеют следующие свойства:

- Функции SFB 36 "NOTIFY" и SFB 31 "NOTIFY\_8P" могут хранить все обнаруженные смены фронта сигнала, если блок вызван, это приводит к передаче сообщения.
- Если процедура создания сообщений имеет установки по умолчанию (выключен запуск сообщений от квитирования), то SFB 33 "ALARM", 34 "ALARM\_8" и 35 "ALARM\_8P" также будут генерировать сообщения, при любом обнаруженном переходе сигнала при вызванном блоке. С другой стороны, если Вы включили запуск сообщений от квитирования, то не все переходы сигнала будут инициировать сообщения (далее этот вопрос будет рассмотрен подробно).
- После выполнения блока связанные (сопутствующие) значения (входы SD\_i) полностью считываются и ставятся в соответствие сообщению (см. раздел "Посылка и прием параметров" в "Общие параметры SFB/FB и SFC/FC для базовой системы связи S7").  
С точки зрения консистентности относительно высокоприоритетных классов следующие сопутствующие значения консистентны:

Каждое связанное значение SD\_i является консистентным.

С помощью параметров состояния DONE, ERROR и STATUS Вы можете отслеживать обработку состояния блока (см. информацию о параметрах состояния в разделе "Общие параметры SFB/FB и SFC/FC для базовой системы связи S7").

---

#### Примечание

Параметры ID и EV\_ID оцениваются только при первом вызове блока (фактические параметры или предварительно определенные значения из экземпляра).

---



### **Регистрация устройств отображения**

Чтобы SFB для создания сообщений, связанных с блоками, могли передавать сообщения при обнаружении смены фронта сигнала, для сообщений, связанных с блоками, должно быть зарегистрировано хотя бы одно устройство отображения. Если нет зарегистрированных устройств отображения, то параметр STATUS = 1.

### **Обнаружение фронта изменения сигнала**

Память для одного сообщения с двумя блоками (memory blocks) доступна для каждого экземпляра (instance) блока сообщений.

Данная память для сообщения первоначально очищается. Смена сигнала на входе SIG или на одном из входов SIG\_1, ..., SIG\_8 вводит сигнал в первый блок памяти (memory block). Этот блок памяти остается занятым, пока соответствующее сообщение не будет переслано.

Следующее изменение сигнала на входе SIG или на одном из входов SIG\_1, ..., SIG\_8 вводится во второй блок памяти. Этот второй блок памяти всегда перезаписывается, если первый блок все еще занят.

Выходные параметры ERROR и STATUS при этом будут индцировать потерю сообщения. Информация о потере сообщения также поступает на интерактивные отображающие устройства, вместе с последующим сообщением, которое может быть переслано.

После того, как первый блок памяти очищается, данные из второго блока памяти передаются в первый блок. При этом второй блок памяти также очищается.

### **Генерация сообщений, запускаемых квитированием**

Вы можете уменьшить поток сообщений в Вашей системе, используя SFB 33 "ALARM", SFB 34 "ALARM\_8" и SFB 35 "ALARM\_8P" с режимом генерации сообщений, запускаемых квитированием.

То есть, после того, как входящее сообщение было первоначально сгенерировано (переход сигнала  $0 > 1$ ), последующие сообщения не будут генерироваться до тех пор, пока Вы не подтвердите первое сообщение с помощью устройства ввода. Следующее сообщение отображается на дисплее после того, как Вы подтвердите уходящее сообщение (переход сигнала  $1 > 0$ ). Цикл сообщений затем возобновится с входящим сообщением (переход сигнала  $0 > 1$ ), которое также должно быть квитировано. Таким образом Вы можете использовать станцию оператора для управления сигналами с сообщениями (сохранение для уходящего сообщения).

Задайте режим функционирования системы сообщений: включите (enable) или выключите (disable) генерацию сообщений, активируемую квитированием - для SFB 33 ... SFB 35 в целом для CPU в Вашей конфигурации в STEP 7. Генерация сообщений, запускаемых квитированием, выключена по умолчанию.

Для обеспечения корректной работы с сообщениями в Вашей системе Вы должны проверить, чтобы все операторские устройства отображения поддерживали квитирование сообщений.

---

**Замечания по поводу операторских устройств отображения, которые не могут управлять сообщениями с помощью квитирования**

CPU с включенным запуском сообщений от квитирования будет распределять сообщения только по операторским устройствам отображения, которые могут управлять сообщениями с помощью квитирования. CPU не будет посылать никаких сообщений, если ни одно из устройств отображения не обладает способностью управлять сообщениями с помощью квитирования. В этом случае выходные параметры ERROR и STATUS имеют значения: ERROR=1 и STATUS=1.

---

**Квитирование сообщений с помощью SFB 33 "ALARM", SFB 34 "ALARM\_8" и SFB 35 "ALARM\_8P"**

Здесь используется концепция централизованного квитирования. Когда Вы подтвердили сообщение на устройстве отображения, информация о квитировании сначала посылается в CPU, генерировавшее это сообщение. Отсюда информация о квитировании распределяется по всем станциям, зарегистрированным для этой цели.

Вы квитируете сигнал, а не отдельное сообщение. Например, если было сообщено о нескольких нарастающих фронтах сигнала, и Вы квитируете поступившее событие, то все предшествующие события с таким же номером сообщения считаются квитированными.

**Отображение квитирования**

SFB36 "NOTIFY" и SFB31 "NOTIFY\_8P" никак не индицирует факт квитирования. Вы можете проверить состояние выходных параметров ACK\_UP и ACK\_DN блока SFB33 "ALARM" и выходных параметров ACK\_STATE блоков SFB 35 "ALARM\_8P" и SFB 34 "ALARM\_8." Эти выходы обновляются в момент вызова блока, если параметр управления EN\_R имеет значение 1.

**Отключение (Disable) и включение (Enable) сообщений посредством SFC или с помощью устройств отображения (WinCC)**

В некоторых случаях может быть полезно подавлять сообщения (например, когда Вы переконфигурируете свою систему). Вы можете отключить (Disable) и включить (Enable) сообщения с помощью устройства отображения или в своей программе. Отключение / включение относится ко всем станциям, зарегистрированным для конкретного сообщения. Сообщение остается выключенным, пока оно вновь не будет включено.

Информацию о заблокированных сообщениях Вы можете получать, используя выходные параметры ERROR и STATUS (ERROR = 1, STATUS = 21).

### Требования к рабочей памяти для SFB для генерации сообщений, связанных с блоками

Для нормальной работы SFB для генерации сообщений, связанных с блоками, требуют для временного использования в рабочей памяти CPU определенных областей, размер которых зависит от данных пользователя (раздел кода).

Размер требуемой памяти показан в следующей таблице:

Тип блока	Требуемый размер памяти в рабочей памяти CPU (в байтах)
NOTIFY	200 + 2 * длина связанного значения, определенного при первом вызове SD_1,...SD_10
NOTIFY_8P	200 + 2 * длина связанного значения, определенного при первом вызове SD_1,...SD_10
ALARM	200 + 2 * длина связанного значения, определенного при первом вызове SD_1,...SD_10
ALARM_8	100
ALARM_8P	200 + 2 * длина связанного значения, определенного при первом вызове SD_1,...SD_10
AR_SEND	54

### Количество передаваемых данных

Данные, передаваемые с помощью связанных значений SD\_i системных функциональных блоков NOTIFY, NOTIFY\_8P, ALARM и ALARM\_8P SFB, не должны превышать установленную максимальную длину.

Максимальная длина данных рассчитывается следующим образом:

Максимальная длина =  $\min(\text{pdu\_local}, \text{pdu\_remote}) - \text{diff} - 4 * (\text{Количество используемых параметров SD}_i)$ ,

где:

- $\min(\text{pdu\_local}, \text{pdu\_remote})$  - это наименьшее значение из массивов данных pdu\_local и pdu\_remote;
- pdu\_local - это максимальная длина массивов данных для локального CPU (см. технические данные Вашего CPU);
- pdu\_remote - это максимальная длина массивов данных для устройств отображения;
- diff = 48, если включен запуск сообщений от квитирования, иначе diff = 44 (если запуск сообщений от квитирования выключен).

**Пример:**

CPU 414-2 посылает сообщения посредством сети Industrial Ethernet в систему WinCC. Запуск сообщений от квитирования выключен.

Используются связанные значения SD\_1, SD\_2 и SD\_3.

pdu\_local = 480 байтов, pdu\_remote = 480 байтов

Количество используемых параметров SD\_i: 3

**Тогда:**

Максимальная длина =  $\min(480, 480) - 44 - 4 * 3 = 480 - 44 - 12 = 424$

Максимальная длина данных, которые могут быть переданы SFB, составляет 424 байта.

## 22.2 Создание с помощью SFB36 "NOTIFY" сообщений, связанных с блоками, без квитирования

### Описание

SFB36 "NOTIFY" контролирует сигнал. Он генерирует сообщение как при нарастающем фронте (состояние прихода события), так и при падающем фронте (состояние ухода события). Вы можете послать с этим сообщением до десяти сопутствующих значений. Это сообщение передается всем зарегистрированным для него станциям. При первом вызове SFB сообщение передается с текущим состоянием сигнала.

Сопутствующие значения опрашиваются при обнаружении фронта и ставятся в соответствие сообщению. Если Вы квитуете сообщение на зарегистрированном устройстве отображения, то об этом сообщается всем остальным зарегистрированным устройствам отображения. Блок NOTIFY не информируется об этом квитировании.

SFB36 "NOTIFY" может временно хранить один нарастающий и один падающий фронт сигнала. Любые последующие изменения сигнала игнорируются. Эта потеря сообщений отображается с помощью выходных параметров ERROR и STATUS (ERROR = 0, STATUS = 11); зарегистрированные устройства отображения также информируются об этой потере. SFB36 "NOTIFY" соответствует стандарту IEC 1131-5.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SIG	INPUT	BOOL	I, Q, M, D, L	Контролируемый сигнал
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE. ID оценивается только при первом вызове.
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения (0 не разрешен). EV_ID оценивается только при первом вызове. После этого при каждом вызове SFB36 с соответствующим экземплярным DB используется номер сообщения из первого вызова. При назначении номеров сообщений используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров сообщений.
SEVERITY	INPUT	WORD	I, Q, M, D, L, константа	Весовой коэффициент события: возможные значения: от 0 до 127 (0 означает наивысший вес)
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE. Генерирование сообщения завершено
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния ERROR
STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS

Параметр	Описание	Тип данных	Область памяти	Характеристика
SD_i, 1 ≤ i ≤ 10	IN_OUT	ANY	I, Q, M, D, T, C	Сопутствующее значение: разрешены следующие типы данных: BOOL (не разрешен битовый массив) BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, P# DB10.DBX5.0 Byte 10).

### Информация об ошибках

Следующая таблица содержит всю информацию об ошибках, относящуюся к SFB36, которая может быть выведена через параметры ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	22	<ul style="list-style-type: none"> <li>Ошибка в указателе на сопутствующие значения SD_i: <ul style="list-style-type: none"> <li>-относительно длины или типа данных</li> <li>-сопутствующие значения в памяти пользователя недоступны, например, из-за DB, который был удален, или из-за ошибки длины области</li> </ul> </li> </ul> Активированное сообщение передается без сопутствующих значений
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи: соединение прервано или не зарегистрировано
1	4	При первом вызове: <ul style="list-style-type: none"> <li>заданный EV_ID находится вне допустимого диапазона или</li> <li>указатель ANY для SD_i имеет формальную ошибку</li> <li>превышен максимально возможный размер памяти, который может быть послан для CPU посредством SFB36</li> </ul>
1	10	Доступ к локальной памяти пользователя невозможен (например, обращение к DB, который был удален)
1	12	При вызове SFB был задан: <ul style="list-style-type: none"> <li>неинициализированный экземпляр DB</li> <li>экземпляр DB, который не принадлежит SFB 36</li> <li>совместно используемый, а не экземпляр DB</li> </ul>
1	18	EV_ID уже используется одним из SFB 33 – 36.
1	20	Недостаточно памяти. H-System: Вызов FSB во время обновления.
1	21	Сообщение с заданным EV_ID заблокировано.

## 22.3 Создание с помощью SFB31 "NOTIFY\_8P" сообщений, связанных с блоками, без отображения квитирования

### Описание

SFB31 "NOTIFY\_8P" представляет собой расширение SFB36 "NOTIFY" до восьми сигналов.

Сообщение генерируется, если был обнаружен по крайней мере один фронт перехода сигнала. Сообщение всегда генерируется в начале вызова SFB31. Все восемь сигналов размещаются под одним общим номером сообщения, который разделяется на восемь "подсообщений" ("sub-message") в устройстве отображения.

Память для одного сообщения с двумя блоками (memory blocks) доступна для каждого экземпляра (instance) SFB31 "NOTIFY\_8P". Для получения информации по промежуточному сохранению переходов сигналов в памяти обратитесь к разделу "Обнаружение фронта переходов сигналов" во введении "Введение в создание сообщений, связанных с блоками, с помощью SFB".

---

### Примечание

Устройство отображения показывает последние два перехода сигнала, независимо от того, были ли потеряны сообщения.

---



---

### Предупреждение

Перед тем, как Вы вызовете SFB31 "NOTIFY\_8P" в CPU, Вы должны обеспечить, чтобы все подключенные устройства отображения "знали" этот блок. Это для случая, если в PLC используются продукты по крайней мере следующих версий: STEP 7 версии V5.1 Service Pack 3, WinCC версии V5.1 Hot Fix 1, PCS 7 версии V5.2 Service Pack 2, SIMATIC Device Driver версии V5.6.

В других случаях коммуникационные соединения PLC с подключенными устройствами отображения будут нарушены. При этом Вы больше не сможете получить доступ к Вашей системе посредством подключенных к ней устройств отображения.

---

Параметры	Объявление	Тип	Область	Описание
SIG_i, 1 ≤ i ≤ 8	INPUT	BOOL	I, Q, M, D, L.	i-й контролируемый сигнал
ID	INPUT	WORD	константа (I, Q, M, D, L)	Канал данных для сообщений: W#16#EEEE_ID проверяется только при первом вызове.
EV_ID	INPUT	DWORD	константа (I, Q, M, D, L)	Номер сообщения (0 не допускается) EV_ID проверяется только при первом вызове. Впоследствии номер сообщения, использованный в первом вызове, используется для каждого вызова SFB31 с соответствующим экземплярным DB. STEP 7 назначает номера сообщений автоматически, чтобы обеспечить их корректность. Номера сообщений внутри пользовательской программы должны быть уникальны.
SEVERITY	INPUT	WORD	константа (I, Q, M, D, L)	Вес события Возможные значения: 0 ... 127 (0 имеет максимальный вес); Значение по умолчанию: 64 Этот параметр не относится к вопросу обработки сообщений
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр статуса DONE: Создание сообщения выполнено.
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметр статуса ERROR ERROR=TRUE показывает, что при обработке произошла ошибка. Детали см. в параметре STATUS.
STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр статуса STATUS: индикатор информации об ошибках
SD_i, 1 ≤ i ≤ 10	IN_OUT	ANY	I, Q, M, D, T, Z	i-е связанное значение Допускаются только данные типов BOOL (недопустимы: битовые массивы), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. <b>Примечание:</b> Когда указатель ANY указывает DB, DB должен быть определен. (например,: P# DB10.DBX5.0 Byte 10)



### Информация об ошибках

В представленной ниже таблице содержится полная информация об ошибках, касающихся SFB31 и которая может быть выведена в параметрах ERROR и STATUS.

Параметр ERROR	Параметр STATUS (десятичный)	Пояснение
0	11	Сообщение потеряно
0	11	Сообщение потеряно, по крайней мере один переход сигнала / одно сообщение не может быть послано
0	22	<ul style="list-style-type: none"> <li>• Ошибка в указателе на связанное значение SD_i:</li> <li>• Ошибка, связанная с длиной / типом данных</li> <li>• Нет доступа к связанному значению в памяти пользователя, например, из-за удаления DB или ошибки длины</li> <li>• Активированное сообщение пересылается без или с максимально возможным номером для связанных сообщений.</li> <li>• Вы задали фактический параметр SEVERITY выше верхнего предела. SEVERITY сообщения получит значение =127.</li> </ul>
0	25	Связь установлена. Сообщение будет обработано.
1	1	Ошибка связи: связь прервана или нет доступа
1	4	<p>В начале вызова</p> <ul style="list-style-type: none"> <li>• параметр EV_ID задан за разрешенными пределами</li> <li>• формальная ошибка указателя ANY на SD_i</li> <li>• превышен максимальный размер данных, который может быть передан CPU с помощью SFB31</li> </ul>
1	10	Нет доступа к локальной памяти пользователя (например, попытка получить доступ к ранее удаленному блоку DB)
1	12	<p>При вызове SFB</p> <ul style="list-style-type: none"> <li>• заданный экземпляр DB не принадлежит SFB31</li> <li>• задан глобальный DB вместо экземплярного DB</li> </ul>
1	18	EV_ID был уже использован одним из SFB31 или SFB33 ... SFB36.
1	20	<p>За пределами рабочей памяти (working memory).</p> <p>H-System: Вызов SFB во время обновления (update) (замены)</p>
1	21	Сообщение с заданным EV_ID заблокировано.

## 22.4 Создание с помощью SFB33 "ALARM" сообщений, связанных с блоками, с квитированием

### Описание

SFB33 "ALARM" контролирует сигнал. Он генерирует сообщение как при нарастающем фронте (приходящее событие), так и при падающем фронте (уходящее событие). С сообщением Вы можете посылать до десяти сопутствующих значений. Сообщение посылается всем зарегистрированным для этого станциям.

При первом вызове SFB сообщение передается с текущим состоянием сигнала.

Выход ACK\_UP сбрасывается, если имеет место нарастающий фронт и генерация сообщения завершена (DONE = 1). Он устанавливается при квитировании вами наступающего события из зарегистрированного устройства отображения. Ситуация для выхода ACK\_DN аналогична: он сбрасывается, когда имеет место падающий фронт и генерирование сообщения завершено (DONE = 1). Он устанавливается при квитировании вами уходящего события из зарегистрированного устройства отображения. Как только Ваше квитирование получено с зарегистрированного устройства отображения, информация о квитировании передается всем зарегистрированным для этого устройствам отображения.

SFB33 "ALARM" может временно хранить один нарастающий и один падающий фронт сигнала. Любые следующие изменения сигнала игнорируются. Эта потеря сообщений отображается с помощью выходных параметров ERROR и STATUS (ERROR = 0, STATUS = 11); зарегистрированные устройства отображения также информируются об этой потере.

SFB33 "ALARM" соответствует стандарту IEC 1131-5.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разблокировано для приема", определяющий, будут ли выходы ACK_UP и ACK_DN обновляться при первом вызове блока (EN_R=1) или нет (EN_R=0). При EN_R=0 SFB33 "ALARM" ведет себя так же, как SFB36 "NOTIFY". Выходные параметры ACK_UP и ACK_DN в этом случае остаются неизменными.
SIG	INPUT	BOOL	I, Q, M, D, L	Контролируемый сигнал
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE. ID оценивается только при первом вызове.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения (0 не разрешен) EV_ID оценивается только при первом вызове. После этого при каждом вызове SFB 33 с соответствующим экземплярным DB используется номер сообщения из первого вызова. При назначении номера сообщения используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров сообщений.
SEVERITY	INPUT	WORD	I, Q, M, D, L, константа	Весовой коэффициент события. Возможные значения: от 0 до 127 (0 означает наибольший вес)
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: генерирование сообщения завершено
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния ERROR
STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS
ACK_DN	OUTPUT	BOOL	I, Q, M, D, L	Уходящее сообщение было квитировано на устройстве отображения
ACK_UP	OUTPUT	BOOL	I, Q, M, D, L	Поступающее сообщение было квитировано на устройстве отображения
SD_i, 1≤i≤10	IN_OUT	ANY	I, Q, M, D, T, C	i-тое связанное значение. Разрешены следующие типы данных: BOOL (не разрешен: битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, P# DB10.DBX5.0 Byte 10).

### Информация об ошибках

Следующая таблица содержит всю информацию об ошибках, относящуюся к SFB33, которая может быть выведена с помощью параметров ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	22	<ul style="list-style-type: none"> <li>Ошибка в указателе на сопутствующие значения SD_i: - Относительно длины или типа данных</li> </ul>

ERROR	STATUS (десятичное число)	Объяснение
		<ul style="list-style-type: none"> <li>- Сопутствующие значения в памяти пользователя недоступны, например, из-за DB, который был удален, или из-за ошибки длины области</li> <li>- Активированное сообщение послано без сопутствующих значений</li> <li>• Выбранный вами фактический параметр для SEVERITY выше допустимого диапазона. Активированное сообщение будет послано с SEVERITY=127.</li> </ul>
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи: соединение прервано или нет регистрации
1	4	<ul style="list-style-type: none"> <li>• При первом вызове: <ul style="list-style-type: none"> <li>- заданный EV_ID находится вне допустимого диапазона или</li> <li>- указатель ANY для SD_i содержит формальную ошибку</li> <li>- превышен максимально возможный размер памяти, который может быть послан для CPU посредством SFB33</li> </ul> </li> </ul>
1	10	Доступ к локальной памяти пользователя невозможен (например, обращение к DB, который был удален)
1	12	При вызове SFB был задан: <ul style="list-style-type: none"> <li>- неинициализированный экземпляр DB</li> <li>- экземпляр DB, не принадлежащий SFB33</li> <li>- совместно используемый, а не экземпляр DB</li> </ul>
1	18	EV_ID уже используется одним из SFB 33 – 36.
1	20	Недостаточно памяти. H-System: Вызов FSB во время обновления.
1	21	Сообщение с заданным EV_ID заблокировано.

## 22.5 Создание сообщений, связанных с блоками, с сопутствующими значениями для восьми сигналов с помощью SFB35 "ALARM\_8P"

### Описание

SFB35 "ALARM\_8P" является прямым расширением SFB33 "ALARM", допускающим восемь сигналов.

Сообщение генерируется, когда обнаруживается смена фронта у одного или более сигналов (исключение: при первом вызове блока сообщение передается всегда). Для всех восьми сигналов существует общий номер сообщения, которое на устройстве отображения разбивается на восемь отдельных сообщений. Вы можете квитирировать каждое отдельное сообщение независимо друг от друга или все восемь отдельных сообщений сразу.

Для включения в свою программу состояния квитирирования для отдельных сообщений Вы можете использовать выходной параметр ACK\_STATE. Если Вы блокируете или разблокируете сообщение блока ALARM\_8P, то это всегда влияет на весь блок ALARM\_8P. Блокировка и разблокировка отдельных сигналов невозможна.

SFB35 "ALARM\_8P" может временно запоминать два сигнала. Любое следующее затем изменение сигнала игнорируется. Эта потеря сообщений отображается выходными параметрами ERROR и STATUS (ERROR = 0, STATUS = 11); зарегистрированные устройства отображения также информируются об этой потере.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разблокировано для приема", который определяет, обновляется ли выход ACK_STATE при вызове блока (EN_R=1) или нет (EN_R=0).
SIG_i, 1 ≤ i ≤ 8	INPUT	BOOL	I, Q, M, D, L	i-й контролируемый сигнал
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE. ID оценивается только при первом вызове.
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения (0 не разрешен). EV_ID оценивается только при первом вызове. После этого при каждом вызове SFB 35 с соответствующим экземплярным DB используется номер сообщения из первого вызова. При назначении номера сообщения используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров сообщений.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SEVERITY	INPUT	WORD	I, Q, M, D, L, константа	Весовой коэффициент события. Возможные значения: от 0 до 127 (0 означает наивысший вес)
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: генерирование сообщения завершено
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния ERROR
STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS
ACK_STATE	OUTPUT	WORD	I, Q, M, D, L	Битовый массив с текущим состоянием квитирования всех восьми сообщений: <ul style="list-style-type: none"> <li>бит <math>2^0</math>: поступающее событие в SIG_1 было квитировано</li> <li>бит <math>2^7</math>: поступающее событие в SIG_8 было квитировано</li> <li>бит <math>2^8</math>: уходящее событие в SIG_1 было квитировано</li> <li>бит <math>2^{15}</math>: уходящее событие в SIG_8 было квитировано</li> </ul>
SD_j, $1 \leq j \leq 10$	IN_OUT	ANY	I, Q, M, D, T, C	j-е сопутствующее значение Эти значения относятся ко всем сообщениям. Разрешены следующие типы данных: BOOL (не разрешен: битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, P# DB10.DBX5.0 Byte 10).

### Информация об ошибках

Следующая таблица содержит всю информацию об ошибках, относящуюся к SFB35, которая может быть выведена с помощью параметров ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	22	<ul style="list-style-type: none"> <li>Ошибка в указателе на сопутствующие значения SD_i: <ul style="list-style-type: none"> <li>- Относительно длины или типа данных</li> <li>- Сопутствующие значения в памяти пользователя недоступны, например, из-за DB, который был удален, или из-за ошибки длины области</li> <li>- Активированное сообщение послано без сопутствующих значений</li> </ul> </li> <li>Выбранный вами фактический параметр для SEVERITY</li> </ul>

ERROR	STATUS (десятичное число)	Объяснение
		выше допустимого диапазона. Активированное сообщение будет послано с SEVERITY=127.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи: соединение прервано или нет регистрации
1	4	При первом вызове: <ul style="list-style-type: none"> <li>заданный EV_ID находится вне допустимого диапазона или</li> <li>указатель ANY для SD_i содержит формальную ошибку</li> <li>превышен максимально возможный размер памяти, который может быть послан для CPU посредством SFB35</li> </ul>
1	10	Доступ к локальной памяти пользователя невозможен (например, обращение к DB, который был удален)
1	12	При вызове SFB был задан: <ul style="list-style-type: none"> <li>неинициализированный экземпляр DB</li> <li>экземпляр DB, не принадлежащий SFB35</li> <li>совместно используемый, а не экземпляр DB</li> </ul>
1	18	EV_ID уже используется одним из SFB 33 – 36.
1	20	Недостаточно памяти. H-System: первый вызов SFB во время обновления данных
1	21	Сообщение с заданным EV_ID заблокировано.

**Примечание**

После первого вызова блока все биты выхода ACK\_STATE установлены, и принимается, что предыдущие значения входов SIG\_i,  $1 \leq i \leq 8$  были равны 0.

## 22.6 Создание сообщений, связанных с блоками, без сопутствующих значений для восьми сигналов с помощью SFB34 "ALARM\_8"

### Описание

SFB34 "ALARM\_8" идентичен SFB35 "ALARM\_8P" за исключением того, что у него нет сопутствующих значений SD\_1 – SD\_10.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN_R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "разблокировано для приема", который определяет, обновляется ли выход ACK_STATE при вызове блока (EN_R=1) или нет (EN_R=0).
SIG_i, 1≤i≤8	INPUT	BOOL	I, Q, M, D, L	i-й контролируемый сигнал
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE. ID оценивается только при первом вызове.
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения (0 не разрешен). EV_ID оценивается только при первом вызове. После этого при каждом вызове SFB34 с соответствующим экземплярным DB используется номер сообщения из первого вызова. При назначении номера сообщения используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров сообщений.
SEVERITY	INPUT	WORD	I, Q, M, D, L, константа	Весовой коэффициент события. Возможные значения: от 0 до 127 (0 означает наивысший вес)
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: генерирование сообщения завершено
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния ERROR
STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS
ACK_STAT E	OUTPUT	WORD	I, Q, M, D, L	Битовый массив с текущим состоянием квитирования всех восьми сообщений: <ul style="list-style-type: none"> <li>бит 2<sup>0</sup>: поступающее событие в SIG_1 было квитировано</li> <li>бит 2<sup>7</sup>: поступающее событие в SIG_8 было квитировано</li> <li>бит 2<sup>8</sup>: уходящее событие в SIG_1 было квитировано</li> <li>бит 2<sup>15</sup>: уходящее событие в SIG_8 было квитировано</li> </ul>



**Информация об ошибках**

Следующая таблица содержит всю информацию об ошибках, относящуюся к SFB34, которая может быть выведена с помощью параметров ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	22	Выбранный вами фактический параметр для SEVERITY выше допустимого диапазона. Активированное сообщение будет послано с SEVERITY = 127.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи: соединение прервано или нет регистрации
1	4	При первом вызове: заданный EV_ID находится вне допустимого диапазона.
1	10	Доступ к локальной памяти пользователя невозможен (например, обращение к DB, который был удален)
1	12	При вызове SFB был задан: <ul style="list-style-type: none"> <li>• неинициализированный экземпляр DB</li> <li>• экземпляр DB, не принадлежащий SFB34</li> <li>• совместно используемый, а не экземпляр DB</li> </ul>
1	18	EV_ID уже используется одним из SFB 33 – 36.
1	20	Недостаточно памяти. H-System: первый вызов SFB во время обновления данных.
1	21	Сообщение с заданным EV_ID заблокировано

**Примечание**

После первого вызова блока все биты выхода ACK\_STATE установлены, и принимается, что предыдущие значения входов SIG<sub>i</sub>, 1 ≤ i ≤ 8 были равны 0.

## 22.7 Передача архивных данных с помощью SFB37 "AR\_SEND"

### Описание

SFB37 "AR\_SEND" передает архивные данные системам взаимодействия с оператором, зарегистрированным для этой цели. Эти системы сообщают CPU о номере соответствующего архива в зарегистрированном сообщении. В зависимости от доступной памяти в CPU и используемой адресной области архивные данные могут иметь длину до 65534 байтов. В структуре архивных данных могут быть приняты во внимание умолчания используемой вами системы взаимодействия с оператором.

Передача данных активируется положительным фронтом на входе управления REQ после вызова блока. Начальный адрес передаваемых архивных данных задается через SD\_1, длина массива данных – через LEN. Передача данных происходит асинхронно по отношению к исполнению программы пользователя. Успешное завершение передачи отображается параметром состояния DONE, имеющим значение 1. Нарастающий фронт на входе управления R прекращает передачу данных.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	Параметр управления "запрос"
R	INPUT	BOOL	I, Q, M, D, L, константа	Параметр управления "сброс": прерывание текущего задания
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE. ID оценивается только при первом вызове.
AR_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер архива (0 не разрешен). AR_ID оценивается только при первом вызове. После этого при каждом вызове SFB 37 с соответствующим экземплярным DB используется номер архива из первого вызова. При назначении номера архива используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров архивов.
DONE	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния DONE: передача завершена
ERROR	OUTPUT	BOOL	I, Q, M, D, L	Параметр состояния ERROR
STATUS	OUTPUT	WORD	I, Q, M, D, L	Параметр состояния STATUS

Параметр	Описание	Тип данных	Область памяти	Характеристика
SD_1	IN_OUT	ANY	I, Q, M, D, T, C	Указатель на архивные данные. Данные о длине не оцениваются. Разрешены следующие типы данных: BOOL (не разрешен: битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. Структура архивных данных должна соответствовать целевой системе. Примечание: Если параметр ANY указывает на DB, то последний всегда должен быть определен (например, R# DB10.DBX5.0 Byte 10).
LEN	IN_OUT	WORD	I, Q, M, D, L	Длина передаваемого массива данных в байтах

### Информация об ошибках

Следующая таблица содержит всю информацию об ошибках, относящуюся к SFB37, которая может быть выведена с помощью параметров ERROR и STATUS.

ERROR	STATUS (десятичное число)	Объяснение
0	11	Предупреждение: новое задание не действует, так как предыдущее задание еще не завершено.
0	25	Обмен данными начался. Задание обрабатывается.
1	1	Проблемы связи
1	2	Отрицательное подтверждение, функция не может быть выполнена
1	3	Нет регистрации для заданного AR_ID.
1	4	<ul style="list-style-type: none"> <li>Ошибка в указателе архивных данных SD_1 в отношении длины данных или типа данных.</li> <li>При первом вызове заданный AR_ID находится вне допустимого диапазона.</li> </ul>
1	5	Было выполнено требование сброса.
1	7	Задание RESET [сброс] не имеет смысла, так как текущая функция завершена или не активирована (блок в неправильном состоянии).
1	10	Доступ к локальной памяти приложения невозможен (например, обращение к DB, который был удален).
1	12	При вызове SFB был задан: <ul style="list-style-type: none"> <li>неинициализированный экземпляр DB</li> <li>экземпляр DB, который не принадлежит SFB37</li> <li>совместно используемый, а не экземпляр DB</li> </ul>
1	18	AR_ID уже использовался блоком SFB37.
1	20	Недостаточно памяти. H-System: вызов SFB во время обновления данных.

## 22.8 Блокировка сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии с помощью SFC10 "DIS\_MSG"

### Описание

С помощью SFC10 "DIS\_MSG" (disable message [блокировать сообщение]) Вы можете блокировать сообщения, связанные с блоком, генерируемые посредством SFB, сообщения, связанные с символом (SCAN), и сообщения о групповом состоянии. Блокируемые сообщения выбирают, используя входные параметры MODE и MESGN. Вызов SFC10 "DIS\_MSG" и успешная блокировка сообщения возможны только тогда, когда блокировка сообщения еще не активирована с помощью SFC10.

Сообщения, готовые к передаче на момент вызова SFC10, но все еще находящиеся во внутреннем буфере, больше не могут блокироваться и передаются. Блокированное сообщение отображается на выходах ERROR и STATUS в SFB "NOTIFY", "ALARM", "ALARM\_8P" и "ALARM\_8".

Блокировку сообщения запускают, присваивая при вызове SFC10 входному параметру REQ значение 1.

### Как функционирует SFC10

Блокировка выполняется асинхронно, другими словами, она может быть активной на протяжении нескольких вызовов SFC10:

- При первом вызове (REQ =1) SFC10 проверяет входные параметры и пытается занять требуемые системные ресурсы. В случае успеха в RET\_VAL вводится значение W#16#7001, устанавливается BUSY и запускается блокировка сообщения. В случае неудачи в RET\_VAL вводится информация об ошибке, и задание завершается. В этом случае BUSY не должен оцениваться.
- Если тем временем имеют место дальнейшие вызовы, то в RET\_VAL вводится значение W#16#7002 (задание все еще выполняется CPU) и устанавливается BUSY. Дальнейшие вызовы не влияют на текущее задание.
- При последнем вызове SFB в RET\_VAL вводится значение W#16#0000, если не было ошибок. Тогда BUSY имеет значение 0. Если произошла ошибка, то в RET\_VAL вводится информация об ошибке и BUSY не должен оцениваться.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: запустить блокировку.
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Параметр для выбора блокируемых сообщений, см. следующую таблицу.
MESGN	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения, существенный только тогда, когда MODE установлен в 5, 6, 7. Это позволяет блокировать отдельное сообщение.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках, см. таблицу "Информация об ошибках".
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: блокировка еще не была отменена.

### Входной параметр MODE

Следующая таблица показывает допустимые значения входного параметра MODE.

Значение	Характеристика
0	Все сообщения CPU, связанные с блоком, связанные с символом, и все сообщения о групповом состоянии, генерируемые с помощью SFB
1	Все связанные с блоком сообщения CPU, генерируемые с помощью SFB, иными словами, все сообщения, генерируемые SFB "NOTIFY", "ALARM", "ALARM_8P" и "ALARM_8"
2	Все сообщения CPU о групповом состоянии
3	Все связанные с символом сообщения CPU (SCAN)
5	Отдельное сообщение класса "сообщения, связанные с символом"
6	Отдельное сообщение класса "сообщения, связанные с блоком"
7	Отдельное сообщение класса "сообщения о групповом состоянии"

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Блокировка завершилась без ошибки.
7000	REQ = 0 при первом вызове: блокировка не была активирована.
7001	REQ = 1 при первом вызове: блокировка была запущена.
7002	Дальнейший вызов: блокировка уже активна.
8081	Ошибка при обращении к параметру.
8082	MODE имеет недопустимое значение.
8083	Номер сообщения находится вне допустимого диапазона значений.
8084	Отсутствует регистрация для сообщения (сообщений), заданного с помощью MODE и, возможно, MESGN.
80C3	Сообщение (сообщения), блокируемое в MODE и, возможно, MESGN, не может блокироваться в настоящее время, так как SFC10 уже блокирует сообщения.

## 22.9 Разблокировка сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии с помощью SFC9 "EN\_MSG"

### Описание

С помощью SFC9 "EN\_MSG" (enable message [разблокировать сообщение]) Вы можете разблокировать сообщения, связанные с блоком, сообщения, связанные с символом, и сообщения о групповом состоянии, которые ранее были заблокированы. Вы блокировали эти сообщения или в устройстве отображения, или с помощью SFC10 "DIS\_MSG".

Сообщения, подлежащие разблокированию, задают, используя входные параметры MODE и MESGN. Успешная разблокировка сообщений с помощью SFC9 "EN\_MSG" возможна только тогда, когда SFC9 уже активно не разблокирует сообщения.

Функцию разблокировки запускают, присваивая при вызове SFC9 входному параметру REQ значение 1.

### Как функционирует SFC9

Разблокировка выполняется асинхронно, другими словами, она может быть активной на протяжении нескольких вызовов SFC9:

- При первом вызове (REQ =1) SFC9 проверяет входные параметры и пытается занять требуемые системные ресурсы. В случае успеха в RET\_VAL вводится значение W#16#7001, устанавливается BUSY и запускается разблокировка сообщения.  
В случае неудачи в RET\_VAL вводится информация об ошибке, и задание завершается. В этом случае BUSY не должен оцениваться.
- Если тем временем имеют место дальнейшие вызовы, то в RET\_VAL вводится значение W#16#7002 (задание все еще выполняется CPU) и устанавливается BUSY. Дальнейшие вызовы не влияют на текущее задание.
- При последнем вызове SFB в RET\_VAL вводится значение W#16#0000, если не было ошибок. Тогда BUSY имеет значение 0. Если произошла ошибка, то в RET\_VAL вводится информация об ошибке и BUSY не должен оцениваться.

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L	REQ = 1: запустить разблокировку.
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Параметр для выбора разблокируемых сообщений, см. следующую таблицу.
MESGN	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения, существенный только тогда, когда MODE установлен в 5, 6, 7. Это позволяет разблокировать отдельное сообщение.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках, см. таблицу.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY = 1: разблокировка еще не была отменена.

### Входной параметр MODE

Следующая таблица показывает допустимые значения входного параметра MODE.

Значение	Характеристика
0	Все сообщения CPU, связанные с блоком, связанные с символом, и все сообщения CPU о групповом состоянии, генерируемые с помощью SFB
1	Все связанные с блоком сообщения CPU, генерируемые с помощью SFB, иными словами, все сообщения, генерируемые SFB "NOTIFY", "ALARM", "ALARM_8P" и "ALARM_8"
2	Все сообщения CPU о групповом состоянии
3	Все связанные с символом сообщения CPU (SCAN)
5	Отдельное сообщение класса "сообщения, связанные с символом"
6	Отдельное сообщение класса "сообщения, связанные с блоком"
7	Отдельное сообщение класса "сообщения о групповом состоянии"

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Разблокировка завершилась без ошибки.
7000	REQ = 0 при первом вызове: разблокировка не была активирована.
7001	REQ = 1 при первом вызове: разблокировка была запущена.
7002	Дальнейший вызов: разблокировка уже активна.
8081	Ошибка при обращении к параметру.
8082	MODE имеет недопустимое значение.
8083	Номер сообщения находится вне допустимого диапазона значений.
8084	Нет регистрации для сообщения (сообщений), заданного с помощью MODE и, возможно, MESGN.
80C3	Сообщение (сообщения), разблокируемое в MODE и, возможно, MESGN, не может быть разблокировано в настоящее время, так как SFC9 уже разблокирует сообщения.

## 22.10 Поведение SFB для создания сообщений, связанных с блоками, при пуске

### Теплый и холодный рестарт

При теплом и холодном рестарте SFB для создания сообщений, связанных с блоками, переводятся в состояние NO\_INIT. Фактические параметры, хранящиеся в экземплярных DB, остаются неизменными.

### Горячий рестарт

При горячем рестарте SFB для создания сообщений, связанных с блоками, ведут себя как функциональные блоки пользователя, способные возобновлять исполнение. Они продолжают выполнение с точки прерывания.

### Сброс памяти

Сброс всегда приводит к разрыву всех соединений, так что станции, зарегистрированные для сообщений, отсутствуют. Программа пользователя удаляется. Если Вы вставили флэш-карту, то существенные для исполнения части программы загружаются оттуда в CPU повторно, и CPU выполняет теплый или холодный рестарт (неявно это всегда холодный рестарт, так как после сброса памяти все данные пользователя инициализируются).



## 22.11 Как SFB для создания сообщений, связанных с блоками, реагируют на неисправности

### Обрыв соединения

Соединения, поставленные в соответствие экземплярам SFB, контролируются на обрыв. При обрыве соединения, затронутые этим станции удаляются в CPU из внутреннего списка станций, зарегистрированных для сообщений, связанных с блоками. Все еще стоящие в очереди сообщения для этих станций удаляются.

Если другие станции все еще зарегистрированы после обрыва соединения, то они продолжают получать сообщения. SFB прекращают передачу сообщений только тогда, когда больше нет соединений ни с одной зарегистрированной станцией. Эту ситуацию отображают выходные параметры ERROR и STATUS (ERROR = 1, STATUS = 1).

### Интерфейс ошибок с программой пользователя

Если при исполнении SFB для создания сообщений, связанных с блоками, появляется ошибка, то SFB переходит в состояние ERROR или ERROR\_E. Одновременно выходной параметр ERROR устанавливается в 1, а в выходной параметр STATUS записывается соответствующий идентификатор ошибки. Эту информацию об ошибке Вы можете оценивать в своей программе.

Примеры возможных ошибок:

- Передача невозможна из-за недостатка ресурсов
- Ошибка при обращении к контролируемым сигналам.

## 22.12 Введение в создание сообщений, связанных с блоками, с помощью SFC

### SFC для создания сообщений, связанных с блоками

Вы можете создать сообщение, связанное с блоком, с помощью следующих SFC:

- SFC17 "ALARM\_SQ"
- SFC18 "ALARM\_S"
- SFC107 "ALARM\_DQ"
- SFC108 "ALARM\_D"

Эти SFC имеют следующие свойства:

- Сообщения, посланные функцией SFC17 "ALARM\_SQ" и SFC107 "ALARM\_DQ", когда состояние сигнала равно 1, могут быть подтверждены на зарегистрированном устройстве отображения. Сообщения SFC18 "ALARM\_S" и SFC108 "ALARM\_D" всегда квитируются неявно. За подробной информацией обратитесь к разделу "Создание квитируемых сообщений, связанных с блоками, с помощью SFC 17 "ALARM\_SQ" и всегда квитируемых сообщений, связанных с блоками, с помощью SFC18 "ALARM\_S".
- После выполнения блока сопутствующее значение SD\_1 считывается целиком и ставится в соответствие сообщению. С точки зрения консистентности относительно классов более высокого приоритета консистентными являются следующие сопутствующие значения:
  - простые типы данных (бит, байт, слово и двойное слово)
  - массив байтового типа, длина которого не превышает максимальной длины, зависящей от типа CPU (см. [/71/](#), [/101/](#)).

### SFC19 "ALARM\_SC"

С помощью SFC 19 "ALARM\_SC" Вы можете запросить:

- состояние квитирования последнего "поступившего сообщения" и состояние сигнала при последнем вызове SFC17 / SFC107 или
- состояние сигнала при последнем вызове SFC18 / SFC108.

### Регистрация устройств отображения

SFC для создания сообщений, связанных с блоками, при вызове передают сообщение только тогда, когда по крайней мере одно устройство отображения зарегистрировано для сообщений, связанных с блоками.

### **Хранение сообщений**

Во избежание потери сообщений при большой загрузке системы связи каждая из SFC 17, 18, 107 и 108 может сохранить в буфере по два сообщения. Если происходит потеря сообщения, то Вы получите информацию об этом посредством RET\_VAL. Зарегистрированные устройства отображения получают информацию об этом при следующей посылке сообщения.

### **Квитирование сообщений в случае SFC17 "ALARM\_SQ" и SFC107 "ALARM\_DQ"**

Если Вы подтвердили в устройстве отображения "факт поступления сообщения", то информация об этом квитировании сначала посылается в CPU, породившее это сообщение. Он затем распределяет информацию о квитировании по всем станциям, зарегистрированным для этой цели.

### **Блокировка и разблокировка сообщений**

Зависимые от блоков сообщения, созданные с помощью SFC 17 "ALARM\_SQ", SFC 18 "ALARM\_S", SFC107 "ALARM\_DQ" или SFC108 "ALARM\_D" не могут быть заблокированы, а затем вновь разблокированы.

### **Обновление сообщений**

Вы можете использовать обновление сообщений на устройстве отображения, чтобы считывать текущее состояние сигнала и квитирования. Во время обновления все зарегистрированные станции продолжают получать сообщения, для которых они зарегистрированы.

---

#### **Примечание**

Когда Вы загружаете блок, уже находящийся в CPU, с помощью вызовов SFC17/SFC18, может оказаться так, что предыдущий блок послал сообщение со статусом "входящее", а новый блок не посылает соответствующего сообщения со статусом "уходящее". Это значит, что сообщение остается во внутренней памяти сообщений CPU. Эта ситуация может возникнуть также, когда Вы удаляете блоки с помощью SFC17/SFC18. Вы можете удалить такие сообщения из внутренней памяти сообщений CPU, переведя CPU в STOP, а затем пройдя через теплый или холодный рестарт.

---

## Изменения в Вашей программе, которая содержит вызовы SFC 17 / SFC 18

Даже если в Вашу программу включены вызовы SFC 107 и/или SFC 108, описанные изменения в программе могут стать причиной того, что сообщения станут резидентом во внутренней памяти для сообщений и, таким образом, займут ресурсы системы.

В отличие от ресурсов системы, занятых вызовами функций SFC 17 / SFC 18, Вы можете освобождать ресурсы системы, занятые вызовами SFC 107/SFC 108, без необходимости переключения CPU в режим STOP. Это выполняется с помощью функции SFC 106 "DEL\_SI" (см. раздел "Освобождение динамически занимаемых ресурсов системы с помощью SFC 106 "DEL\_SI"). Прежде чем Вы освободите динамически занимаемые ресурсы системы вызовом SFC 106 "DEL\_SI", имеет смысл посмотреть информацию о текущем состоянии ресурсов CPU с помощью функции SFC 105 "READ\_SI", см. раздел "Считывание состояния динамически занимаемые ресурсы системы с помощью SFC 105 "READ\_SI".

**Также обратитесь к разделу:**

"Конфигурирование сообщений"

## Допустимое количество передаваемых данных

Количество данных, передаваемое с использованием значения SD для функций SFC ALARM\_S, ALARM\_SQ, ALARM\_D и ALARM\_DQ не может превышать некоторой максимальной длины, вычисляемой следующим образом:

$\text{Maxleng} = \min(\text{pdu\_local}, \text{pdu\_remote}) - 48$

Формирования рисунка(четкости):

- pdu\_local: максимальная длина блоков данных в CPU (SZL\_ID W#16#0131, INDEX 1, Variable pdu)
- pdu\_remote: максимальная длина блоков данных устройства отображения

### Пример:

CPU 414-1 посылает сообщение в программатор PG 760 (через MPI).

pdu\_local = 480 байтов, pdu\_remote = 480 байтов,

Результат:

$\text{Maxleng} = \min(480, 480) - 48 = 480 - 48 = 432$

Ответ: максимальная длина данных для передачи составляет 432 байта.

## 22.13 Создание квитируемых сообщений, связанных с блоками, с помощью SFC 17 "ALARM\_SQ" и всегда квитируемых сообщений, связанных с блоками, с помощью SFC18 "ALARM\_S"

---

### Примечание

Во вновь создаваемых программах должны использоваться только SFC 107 и SFC 108, так как они обеспечивают улучшенное управление системными ресурсами системы.

---

### Описание

SFC 17 "ALARM\_SQ" и SFC 18 "ALARM\_S" при каждом вызове генерируют сообщение, к которому Вы можете добавить сопутствующие значения. Это сообщение передается всем зарегистрированным для станциям. SFC17 и SFC18 предоставляют в Ваше распоряжение простой механизм для передачи сообщений. Вы должны обеспечить вызов SFC17 или SFC18 только тогда, когда значение запускающего сигнала SIG инвертируется по сравнению с последним вызовом. Если этого не происходит, то это отображается в RET\_VAL, и сообщение не посылаются. При самом первом вызове SFC17 или SFC18 Вы должны позаботиться о том, чтобы на входе SIG была 1. Иначе RET\_VAL будет содержать информацию об ошибке, и сообщение не будет передано.

---

### Примечание

Вызывайте SFC 17 и SFC 18 из FB, которому Вы предварительно назначили соответствующие системные атрибуты! Более подробную информацию по назначению блокам системных атрибутов Вы найдете в [/232/](#) и [/233/](#).

---

### Использование системных ресурсов

При генерации сообщений с помощью SFC17 "ALARM\_SQ" и SFC18 "ALARM\_S" операционная система использует один системный ресурс (экземпляр) на протяжении цикла сигнала. Для SFC18 "ALARM\_S" цикл сигнала длится от вызова SFC с SIG = 1 до вызова SFC с SIG = 0. Для SFC17 "ALARM\_SQ" этот период времени также ограничен промежутком до момента, пока входящий сигнал не будет подтвержден одним из устройств отображения, если это необходимо.

Если во время цикла сигнала блок генерации сообщения был перезагружен или удален, связанный системный ресурс останется занятым до следующего перезапуска (теплого перезапуска).

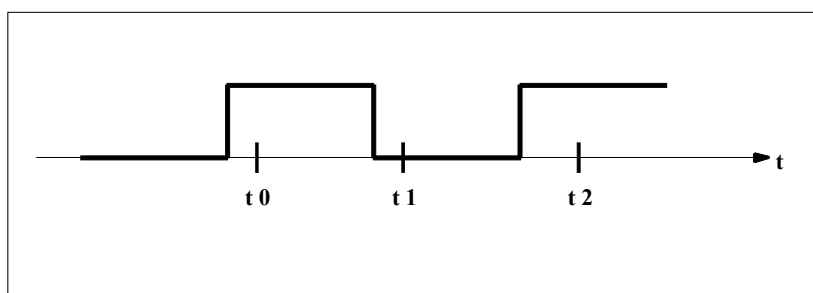
### Квитирование сообщений

Вы можете квитировать сообщения, передаваемые SFC17 "ALARM\_SQ", когда состояние контролируемого сигнала равно 1. Вы можете опросить состояние квитирования последнего "поступившего" сообщения и состояние сигнала при последнем вызове SFC с помощью SFC19 "ALARM\_SC". Сообщения, посланные с помощью SFC18 "ALARM\_S", всегда неявно квитируются. Вы можете опросить состояние сигнала при последнем вызове SFC18 с помощью SFC19 "ALARM\_SC".

### Временное запоминание состояний сигналов

SFC17 "ALARM\_SQ" и SFC18 "ALARM\_S" временно занимают память. Сюда они вводят, среди прочего, последние два состояния сигнала, включая отметку времени и сопутствующее значение. Если SFC17 или SFC18 вызывается в момент времени, когда состояния сигнала двух последних "имеющих силу" вызовов SFC еще не переданы (переполнение буфера сигналов), то текущее и последнее состояние сигнала отбрасываются, и в буферной памяти устанавливается признак переполнения. В ближайший возможный момент времени передается предпоследний сигнал вместе с признаком переполнения.

Пример:



$t_0$ ,  $t_1$  и  $t_2$  – это моменты времени, когда вызываются SFC17 или SFC18. Если состояния сигнала в моменты  $t_0$  и  $t_1$  еще не переданы к моменту времени  $t_2$ , то состояния сигнала, относящиеся к моментам  $t_1$  и  $t_2$ , отбрасываются, а для состояния сигнала, соответствующего моменту  $t_0$  устанавливается признак переполнения.

### Переполнение экземпляров

Если количество вызовов SFC17 или SFC18 больше, чем максимальное количество динамических экземпляров, то результатом этого может быть недостаток ресурсов (переполнение экземпляров). Об этом сообщается как посредством информации об ошибке в RET\_VAL, так и индикацией на зарегистрированных устройствах отображения.

Максимальное количество вызовов SFC17 или SFC18 зависит от CPU. Эту информацию Вы можете найти в [/70/](#) и [/101/](#).

Параметр	Описание	Тип данных	Область памяти	Характеристика
SIG	INPUT	BOOL	I, Q, M, D, L	Сигнал для запуска сообщения
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения (0 не разрешен). При назначении номера сообщения используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров сообщений.
SD	INPUT	ANY	I, Q, M, D, T, C	Сопутствующее значение. Максимальная длина: 12 байтов Разрешены следующие типы данных: BOOL (не разрешен: битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
0001	<ul style="list-style-type: none"> <li>Сопутствующее значение длиннее, чем максимально допустимая длина, или</li> <li>Доступ к памяти пользователя невозможен (например, обращение к ранее удаленному DB). Сообщение передается.</li> </ul>
0002	Предупреждение: была использована последняя свободная память для квитирования сообщений.
8081	Заданный EV_ID находится вне области допустимых значений.
8082	Потеря сообщений, так как Ваш CPU не имеет больше ресурсов для создания сообщений, связанных с блоками, с помощью SFC.
8083	Потеря сообщения, так как та же самая смена сигнала уже имеется, но еще не смогла быть передана (переполнение буфера сигналов).
8084	Сигнал SIG, запустивший сообщение, при текущем и при предыдущем вызове SFC17 или SFC18 имеет одно и то же значение.
8085	Отсутствует регистрация для указанного EV_ID
8086	Вызов SFC для заданного EV_ID уже выполняется классе более низкого приоритета.
8087	При первом вызове SFC17 или SFC18 сигнал запуска сообщения имел значение 0.
8088	Заданный EV_ID уже используется SFC другого типа, который в данный момент времени (все еще) занимает память. Заданный EV_ID уже используется SFC другого типа, который в данный момент времени (все еще) занимает память.

## 22.14 Определение состояния квитирования последнего сообщения для наступающего события ALARM\_SQ/ALARM\_DQ с помощью SFC19 "ALARM\_SC"

### Описание

С помощью SFC 19 "ALARM\_SC" Вы можете опросить:

- состояние квитирования последнего сообщения для наступающего события ALARM\_SQ/ALARM\_DQ и состояние сигнала, запустившего это сообщение при последнем вызове SFC17 "ALARM\_SQ"/SFC107 "ALARM\_DQ".
- состояние сигнала, запустившего сообщение при последнем вызове SFC18 "ALARM\_S"/SFC108 "ALARM\_D".

Если Вы назначили номера сообщений при их конфигурировании, то обращение к сообщению или сигналу происходит с помощью уникального номера сообщения. SFC19 "ALARM\_SC" получает доступ к временно занятой памяти функции SFC17, SFC18, SFC107 или SFC108.

Параметр	Описание	Тип данных	Область памяти	Характеристика
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения, для которого Вы хотите определить состояние сигнала при последнем вызове SFC или состояние квитирования последнего "поступившего" сообщения (только для SFC17 и SFC107).
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке
STATE	OUTPUT	BOOL	I, Q, M, D, L	Состояние сигнала, запустившего сообщение, при последнем вызове SFC
Q_STATE	OUTPUT	BOOL	I, Q, M, D, L	Если заданный параметр EV_ID относится к вызову SFC18/ SFC108: 1 Если заданный параметр EV_ID относится к вызову SFC 17/ SFC 107: состояние квитирования последнего "поступившего" сообщения: 0: не квитировано; 1: квитировано

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
8081	Заданный EV_ID находится вне допустимой области.
8082	Для этого EV_ID в данный момент времени не зарезервировано место в памяти (возможная причина: соответствующий сигнал еще не имел состояния 1 или он уже опять принял состояние 0).



## 22.15 Создание квитируемых сообщений, связанных с блоками, с помощью SFC 107 "ALARM\_DQ" и всегда квитируемых сообщений, связанных с блоками, с помощью SFC108 "ALARM\_D"

### Описание

SFC 107 "ALARM\_DQ" и SFC 108 "ALARM\_D" при каждом вызове генерируют сообщение, к которому Вы можете добавить сопутствующие значения так же как и с SFC 17 "ALARM\_SQ" и 18 "ALARM\_S".

При производстве сообщений с SFC 107 "ALARM\_DQ" и 108 "ALARM\_D", операционная система временно занимает ресурс системы на протяжении цикла.

Для SFC 108 "ALARM\_D" цикл сигнала определяется промежутком времени от вызова SFC с параметром SIG = 1 до другого вызова с параметром SIG = 0. Для SFC 107 "ALARM\_DQ", этот интервал времени также включает промежуток времени до появления входящего сигнала, подтверждающего сообщение на одном из устройств отображения, если это необходимо.

Если во время цикла генерирующий сообщение блок будет перезагружен или удален, связанные системные ресурсы останутся занятыми до следующего теплого рестарта.

Дополнительные функциональные возможности SFC 107 "ALARM\_DQ" и 108 "ALARM\_D" в отличие от SFC 17 и 18 заключаются в том, что Вы имеете возможность управлять этими занятыми ресурсами системы:

- С помощью SFC 105 "READ\_SI" Вы можете выбирать информацию в занятых ресурсах системы.
- С SFC 106 "DEL\_SI" Вы можете освобождать занятые ресурсы системы. Это имеет особое значение для постоянно занимаемых ресурсов системы. Текущий занятый ресурс системы, например, остается занятым до следующего теплого рестарта, если Вы, в ходе изменения программы, удаляете вызов FB, который содержит вызовы SFC108 или SFC107. Если Вы измените программу и перезагрузите вызов FB SFC 107 или SFC 108, то может случаться, что SFC 107 и 108 больше не генерируют сообщения.

SFC 107 и 108 содержат на один параметр больше чем SFC 17 и 18, а именно входной параметр CMP\_ID. Используйте этот параметр, чтобы назначить сообщения, генерируемые SFC 107 и 108 в логические области, например, в системную область памяти. Если Вы вызываете SFC 107/SFC 108 в FB обычно необходимо указать номер соответствующего экземпляра DB для CMP\_ID.

Параметр	Описание	Тип данных	Область памяти	Характеристика
SIG	INPUT	BOOL	I, Q, M, D, L	Сигнал для запуска сообщения
ID	INPUT	WORD	I, Q, M, D, L, константа	Канал данных для сообщений: W#16#EEEE
EV_ID	INPUT	DWORD	I, Q, M, D, L, константа	Номер сообщения (0 не разрешен). При назначении номера сообщения используйте функции конфигурирования сообщений. Это гарантирует консистентность номеров сообщений.
CMP_ID	INPUT	DWORD	I, Q, M, D, L, константа	Идентификатор компонента (0 не разрешен). Идентификатор части системы, для которого предназначается сообщение. Рекомендуемые значения: • младшее слово: 1 ... 65535 • старшее слово: 0 Использование этих рекомендаций гарантирует корректность использования Вами пакета программ SIEMENS.
SD	INPUT	ANY	I, Q, M, D, T, C	Сопутствующее значение. Максимальная длина: 12 байтов Разрешены следующие типы данных: BOOL (не разрешен: битовый массив), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибке

### Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Ошибок не было.
0001	<ul style="list-style-type: none"> <li>Сопутствующее значение длиннее, чем максимально допустимая длина, или</li> <li>Доступ к памяти пользователя невозможен (например, обращение к ранее удаленному DB). Сообщение передается.</li> </ul>
0002	Предупреждение: была использована последняя свободная область памяти для квитиования сообщений.
8081	Заданный EV_ID находится вне области допустимых значений.
8082	Потеря сообщений, так как Ваш CPU не имеет больше ресурсов для создания сообщений, связанных с блоками, с помощью SFC.

Код ошибки (W#16#...)	Объяснение
8084	Сигнал SIG, запустивший сообщение, при текущем и при предыдущем вызове SFC107 или SFC108 имеет одно и то же значение.
8085	Отсутствует регистрация для указанного EV_ID
8086	Вызов SFC для заданного EV_ID уже выполняется в классе более низкого приоритета.
8087	При первом вызове SFC107 или SFC108 сигнал запуска сообщения имел значение 0.
8088	Заданный EV_ID уже используется SFC другого типа, который в данный момент времени (все еще) занимает память.
8089	Вы назначили значение 0 параметру CMP_ID.

## 22.16 Считывание данных из динамически занимаемых областей системной памяти с помощью SFC105 "READ\_SI"

### Как динамически занятые ресурсы системы используются при формировании сообщений с помощью SFC 107 и 108

Когда сообщения сгенерированы с помощью SFC 107 "ALARM\_DQ" и 108 "ALARM\_D", операционная система временно занимает часть памяти системы.

Например, если Вы не удаляете FB, который существует в CPU с вызовами функций SFC 107 или SFC 108, может случиться, что соответствующие ресурсы системы останутся постоянно заняты. Если Вы перезагрузите FB с вызовами SFC 107 или SFC 108, то может случиться, что SFC 107 и 108 не будут больше корректно обрабатываться.

#### Описание

С помощью системной функции SFC105 "READ\_SI" Вы можете считать используемые в настоящее время системные ресурсы, занятые SFC 107 и SFC 108 при генерации сообщений. Это выполняется с помощью значений параметров EV\_ID и CMP\_ID, используемых здесь. Эти значения передаются в SFC105 "READ\_SI" в параметре SI\_ID.

SFC105 "READ\_SI" имеет 4 возможных рабочих режима, которые поясняются ниже в таблице. Выбирайте требуемый режим посредством параметра MODE.

MODE (Режим)	Какой из системных ресурсов, занятых SFC107/SFC108, требуется считывать?
1	Все (вызов SFC105 с параметром SI_ID:=0)
2	Системный ресурс, занятый вызовом SFC107-/SFC108 с EV_ID:=ev_id (вызов SFC105 с параметром SI_ID:=ev_id)
3	Системный ресурс, занятый вызовом SFC107-/SFC108 с CMP_ID:=cmp_id (вызов SFC105 с параметром SI_ID:=ev_id)
0	Дополнительные системные ресурсы, которые не могут быть считаны при предшествующем вызове с MODE=1 или MODE=3, потому что слишком малое поле назначения SYS_INST определено пользователем

### Принцип использования

Если Вы выбрали достаточно большую целевую область SYS\_INST (область назначения) при вызове SFC105 в режимах MODE=1 или MODE=3, в ней будет находиться содержимое всех в настоящее время занятых системных ресурсов, выбранных посредством параметра MODE.

Большая загрузка системных ресурсов приводит к соответствующему высокому значению времени выполнения (runtime) SFC. То есть, большая загрузка CPU может привести к превышению сконфигурированного максимального времени мониторинга цикла (cycle monitoring time).

Вы можете решить эту проблему следующим образом: Выберите относительно малую целевую область SYS\_INST. Значение параметра RET\_VAL=W#16#0001 информирует Вас о том, что SFC не может ввести все системные ресурсы, которые необходимы в область SYS\_INST. В этом случае вызывайте SFC105 в режиме MODE=0 и с тем же SI\_ID, что и в предыдущий раз, пока значение RET\_VAL равно W#16#0000.

---

#### Примечание

Так как операционная система не координирует вызовы SFC105, относящиеся к заданию считывания, Вы должны выполнять все вызовы SFC105 в одном приоритетном классе.

---

### Структура целевой области SYS\_INST

Целевая область для размещения считываемых системных ресурсов должны располагаться внутри DB. Вы должны определить целевую область соответствующим образом, как область структур, посредством чего создается структура в соответствии с таблицей:

Элемент структуры	Тип данных	Описание
SFC_NO	WORD	Номер функции SFC, которая занимает системный ресурс
LEN	BYTE	Длина структуры в байтах, включая SFC_NO и LEN: B#16#0C
SIG_STAT	BOOL	Состояние сигнала (Signal state)
ACK_STAT	BOOL	Статус квитирования (Acknowledgement status) приходящего события (положительный фронт)
EV_ID	DWORD	Номер сообщения
CMP_ID	DWORD	Частный системный идентификатор (Partial system ID)

Параметры	Объявление	Тип	Область	Описание
MODE	INPUT	INT	I, Q, M, D, L, константа	Идентификатор задания Допустимые значения <ul style="list-style-type: none"> <li>• 1: Считать все сист. ресурсы</li> <li>• 2: Считать сист. ресурсы, которые заняты с EV_ID = ev_id при вызове SFC107-/SFC108</li> <li>• 3: Считать сист. ресурсы, которые заняты с CMP_ID:=cmp_id при вызове SFC107-/SFC108</li> <li>• 0: последующий вызов</li> </ul>
SI_ID	INPUT	DWORD	I, Q, M, D, L, константа	ID сист. ресурсов, которые должны быть считаны Допустимые значения <ul style="list-style-type: none"> <li>• 0, если MODE=1</li> <li>• номер сообщ. ev_id, если MODE=2</li> <li>• ID cmp_id для идентификации раздела системы, если MODE=3</li> </ul>
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Возвращаемое значение (статус задания или информация об ошибках)
N_SI	OUTPUT	INT	I, Q, M, D, L	Число выводимых сист. ресурсов с помощью SYS_INT
SYS_INT	OUTPUT	ANY	D	Целевая область для считывания системных ресурсов

### Информация об ошибках

Код ошибки (W#16#...):	Расшифровка
0000	Ошибок нет
0001	Не все системные ресурсы могут быть считаны, так как заданная целевая область SYS_INT слишком мала.
8081	(только для MODE=2 или 3) Вы назначили значение 0 для SI_ID.
8082	(только для MODE=1) Вы назначили значение 0 для одного из SI_ID.
8083	(только для MODE=0) Вы назначили SI_ID другое значение, не такое как в предыдущем вызове SFC с параметром MODE=1 или 3.
8084	Вы назначили некорректное значение для параметра MODE.
8085	SFC105 уже выполняется в другом ОБ.
8086	Целевая область SYS_INT слишком мала для системного ресурса.
8087	Целевая область SYS_INT не существует в DB.
8хху	Общая информация об ошибках, см. <i>Проверка ошибок с помощью выходного параметра RET_VAL</i>

## 22.17 Удаление данных из динамически занимаемых областей системной памяти с помощью SFC106 "DEL\_SI"

### Как динамически занятые ресурсы системы используются при формировании сообщений с помощью SFC 107 и 108

Когда сообщения сгенерированы с помощью SFC 107 "ALARM\_DQ" и 108 "ALARM\_D", операционная система временно занимает часть памяти системы.

Например, если Вы не удаляете FB, который существует в CPU с вызовами функций SFC 107 или SFC 108, может случиться, что соответствующие ресурсы системы останутся постоянно заняты. Если Вы перезагрузите FB с вызовами SFC 107 или SFC 108, то может случиться, что SFC 107 и 108 не будут больше корректно обрабатываться.

### Описание

С помощью системной функции SFC 106 "DEL\_SI" Вы можете освобождать системные ресурсы, занятые в настоящее время.

SFC 106 "DEL\_SI" имеет три возможных рабочих режима, которые поясняются ниже в таблице. Установите требуемый рабочий режим посредством параметра MODE.

MODE (Режим)	Какой из системных ресурсов, занятых SFC107/SFC108, требуется освободить?
1	Все (вызов SFC106 с параметром SI_ID:=0)
2	Системный ресурс, занятый вызовом SFC107-/SFC108 с EV_ID:=ev_id (вызов SFC106 с параметром SI_ID:=ev_id)
3	Системный ресурс, занятый вызовом SFC107-/SFC108 с CMP_ID:=cmp_id (вызов SFC106 с параметром SI_ID:=ev_id)

Параметры	Объявление	Тип	Область	Описание
MODE	INPUT	INT	I, Q, M, D, L, константа	Идентификатор задания Допустимые значения <ul style="list-style-type: none"> <li>1: Освободить все сист. ресурсы</li> <li>2. Освободить сист. ресурсы, которые заняты с EV_ID = ev_id при вызове SFC107-/SFC108</li> <li>3: Освободить сист. ресурсы, которые заняты с CMP_ID:=cmp_id при вызове SFC107-/SFC108</li> </ul>
SI_ID	INPUT	DWORD	I, Q, M, D, L, константа	ID сист. ресурсов, которые должны быть освобождены Допустимые значения <ul style="list-style-type: none"> <li>0, если MODE=1</li> <li>номер сообщ. ev_id, если MODE=2</li> <li>ID cmp_id для идентификации раздела системы, если MODE=3</li> </ul>
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Информация об ошибках

### Информация об ошибках

Код ошибки (W#16#...):	Расшифровка
0000	Ошибок не было.
8081	(только для MODE=2 или 3) Вы назначили значение 0 для SI_ID.
8082	(только для MODE=1) Вы назначили значение 0 для одного из SI_ID.
8084	Вы назначили некорректное значение для параметра MODE.
8085	SFC106 уже выполняется.
8086	Не все выбранные системные ресурсы могут быть удалены, так как по крайней мере один из них был обрабатывался во время вызова SFC106.
8хху	Общая информация об ошибках, см. <i>Проверка ошибок с помощью выходного параметра RET_VAL</i>



## **23 Таймеры и счетчики IEC**

## 23.1 Генерирование импульса с помощью SFB3 "TP"

### Описание

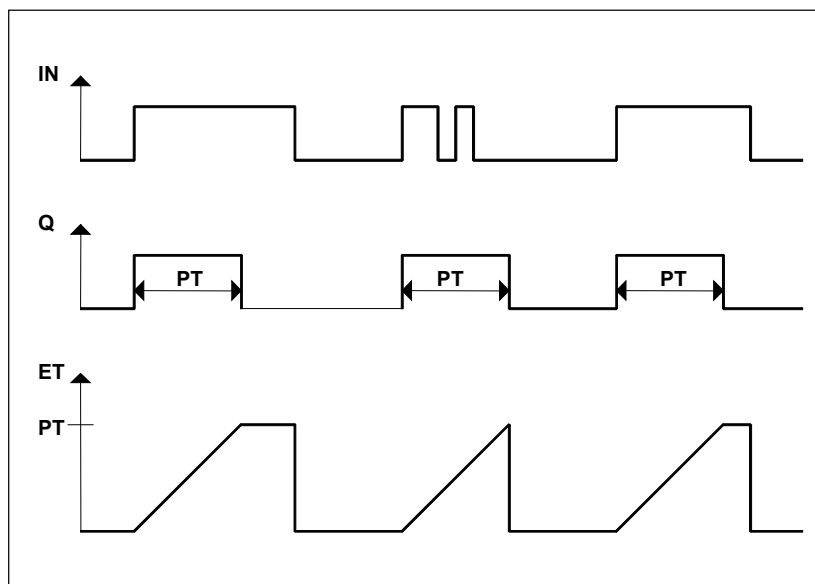
SFB3 "TP" генерирует импульс длительностью PT. Таймер работает только в режимах STARTUP (запуск) и RUN (выполнение).

Нарастающий фронт сигнала на входе IN запускает импульс. Выход Q остается установленным в течение промежутка времени PT, независимо от изменений во входном сигнале (иными словами, даже тогда, когда вход IN вновь меняет состояние с 0 на 1 до истечения времени PT). Выход ET дает информацию о времени, в течение которого выход Q уже является установленным. Максимальное значение выхода ET равно значению входа PT. Выход ET сбрасывается, когда вход IN изменяет свое состояние на 0, но не прежде, чем истечет время PT.

SFB3 "TP" соответствует стандарту IEC 1131-3.

Операционная система сбрасывает экземпляры SFB3 "TP" во время холодного рестарта. Если Вы хотите, чтобы экземпляры этого SFB были инициализированы после теплого рестарта, Вы должны вызвать SFB3 с  $PT = 0$  мс в OB100. Если экземпляры этого SFB расположены внутри другого блока, Вы можете сбросить эти экземпляры, например, инициализацией блока более высокого уровня.

### Временная диаграмма



Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	BOOL	I, Q, M, D, L, константа	Вход запуска
PT	INPUT	TIME	I, Q, M, D, L, константа	Длительность импульса. PT должен быть положительным. (Примечание: диапазон значений фиксирован типом данных TIME)
Q	OUTPUT	BOOL	I, Q, M, D, L	Состояние времени
ET	OUTPUT	TIME	I, Q, M, D, L	Истекшее время

## 23.2 Генерирование задержки включения с помощью SFB4 "TON"

### Описание

SFB4 "TON" задерживает нарастающий фронт сигнала на время PT. Таймер работает только в режимах STARTUP (запуск) и RUN (выполнение).

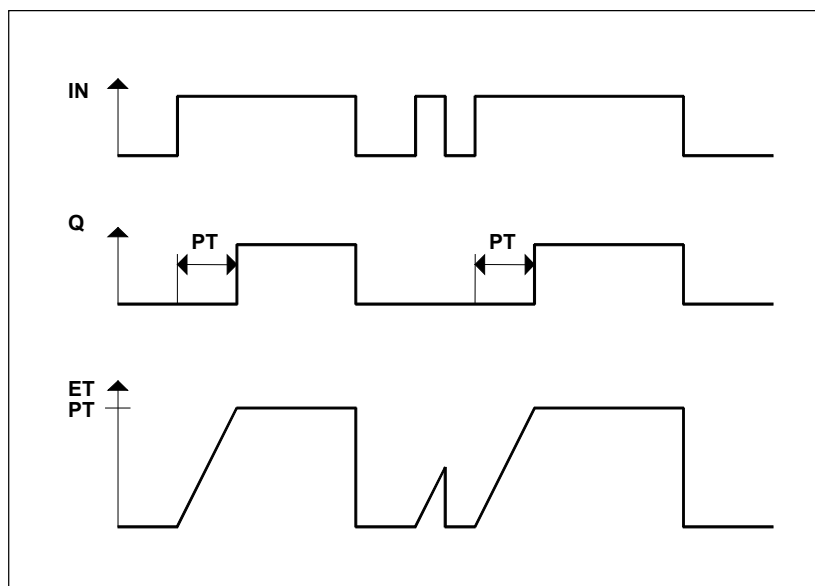
Нарастающий фронт на входе IN вызывает нарастающий фронт на выходе Q по истечении промежутка времени PT. Потом Q остается установленным до тех пор, пока вход IN не сменит состояние на 0. Если вход IN изменяет состояние на 0 до того, как истекло время PT, то выход Q остается установленным на 0.

Выход ET дает информацию о времени, которое прошло с момента последнего нарастающего фронта на входе IN. Его максимальное значение равно значению входа PT. Выход ET сбрасывается, когда вход IN изменяет состояние на 0.

SFB4 "TON" соответствует стандарту IEC 1131-3.

Операционная система сбрасывает экземпляры SFB4 "TON" во время холодного рестарта. Если Вы хотите, чтобы экземпляры этого SFB были инициализированы после теплого рестарта, Вы должны вызвать SFB4 с  $PT = 0$  мс в OB100. Если экземпляры этого SFB расположены внутри другого блока, то Вы можете сбросить эти экземпляры, например, инициализацией блока более высокого уровня.

### Временная диаграмма



Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	BOOL	I, Q, M, D, L, константа	Вход запуска
PT	INPUT	TIME	I, Q, M, D, L, константа	Промежуток времени, на который задерживается нарастающий фронт на входе IN. PT должен быть положительным. (Примечание: диапазон значений фиксирован типом данных TIME)
Q	OUTPUT	BOOL	I, Q, M, D, L	Состояние времени
ET	OUTPUT	TIME	I, Q, M, D, L	Истекшее время

## 23.3 Генерирование задержки выключения с помощью SFB5 "TOF"

### Описание

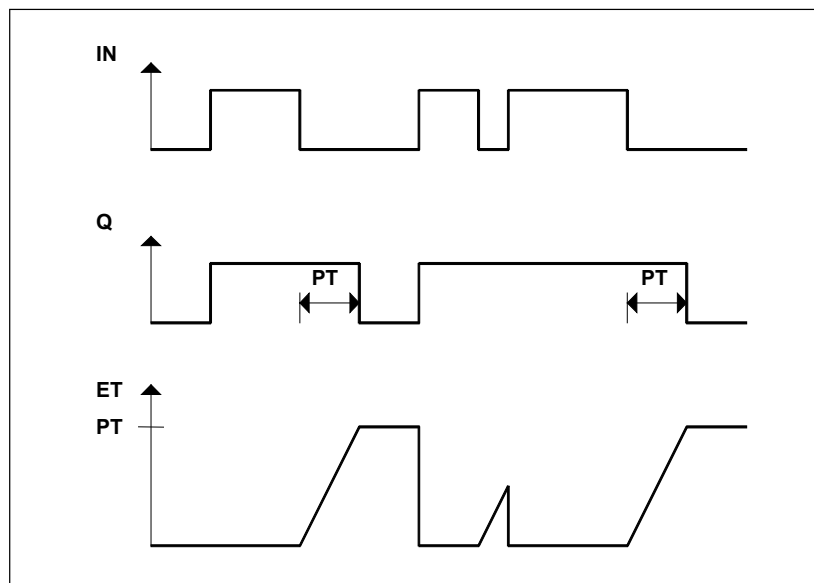
SFB5 "TOF" задерживает падающий фронт на время PT. Таймер работает только в режимах STARTUP (запуск) и RUN (выполнение).

Нарастающий фронт на входе IN вызывает нарастающий фронт на выходе Q. Падающий фронт на входе IN вызывает падающий фронт на выходе Q по истечении промежутка времени PT. Если вход IN вновь изменяет состояние на 1 до того, как истекло время PT, то выход Q остается установленным в 1. Выход ET дает информацию о времени, истекшем после последнего падающего фронта на входе IN. Однако его максимальное значение равно значению входа PT. ET сбрасывается, когда вход IN изменяет свое состояние на 1.

SFB5 "TOF" соответствует стандарту IEC 1131-3.

Операционная система сбрасывает экземпляры SFB5 "TOF" во время холодного рестарта. Если Вы хотите, чтобы экземпляры этого SFB были инициализированы после теплого рестарта, Вы должны вызвать SFB5 с  $PT = 0$  мс в OB100. Если экземпляры этого SFB расположены внутри другого блока, то Вы можете сбросить эти экземпляры, например, инициализацией блока более высокого уровня

### Временная диаграмма



Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	BOOL	I, Q, M, D, L, константа	Вход запуска
PT	INPUT	TIME	I, Q, M, D, L, константа	Промежуток времени, на который задерживается падающий фронт на входе IN. PT должен быть положительным. (Примечание: диапазон значений фиксирован типом данных TIME)
Q	OUTPUT	BOOL	I, Q, M, D, L	Состояние времени
ET	OUTPUT	TIME	I, Q, M, D, L	Истекшее время

## 23.4 Прямой счет с помощью SFB0 "CTU"

### Описание

С помощью SFB 0 "CTU" Вы можете вести прямой счет. Счетчик увеличивается на 1 при нарастающем фронте на входе CU (по сравнению с последним вызовом SFB). Если значение счетчика достигает верхней границы 32 767, то оно больше не увеличивается. После этого каждый следующий нарастающий фронт на входе CU остается без последствий.

Уровень сигнала 1 на входе R вызывает сброс счетчика в 0, независимо от того, какое значение имеет место на входе CU.

На выходе Q отображается, является ли текущее значение счетчика большим или равным предварительно заданному значению PV.

SFB0 "CTU" соответствует стандарту IEC 1131-3.

Операционная система сбрасывает экземпляры SFB0 "CTU" во время холодного рестарта. Если Вы хотите, чтобы экземпляры этого SFB были инициализированы после теплого рестарта, Вы должны вызвать SFB0 с R = 1 в OB100. Если экземпляры этого SFB расположены внутри другого блока, то Вы можете сбросить эти экземпляры, например, инициализацией блока более высокого уровня

Параметр	Описание	Тип данных	Область памяти	Характеристика
CU	INPUT	BOOL	I, Q, M, D, L, константа	Вход счетчика.
R	INPUT	BOOL	I, Q, M, D, L, константа	Вход сброса R имеет преимущество перед CU.
PV	INPUT	INT	I, Q, M, D, L, константа	Предварительно заданное значение. О воздействии PV см. параметр Q.
Q	OUTPUT	BOOL	I, Q, M, D, L	Состояние счетчика: Q имеет следующее значение <ul style="list-style-type: none"> <li>• 1, если <math>CV \geq PV</math></li> <li>• 0 в противном случае</li> </ul>
CV	OUTPUT	INT	I, Q, M, D, L	Текущее значение счетчика. (Возможные значения: от 0 до 32 767).



## 23.5 Обратный счет с помощью SFB1 "CTD"

### Описание

С помощью SFB 1 "CTD" Вы можете выполнять обратный счет. Счетчик уменьшается на 1 при нарастающем фронте на входе CD (по сравнению с последним вызовом SFB). Если значение счетчика достигает нижней границы -32 767, то оно больше не уменьшается. После этого каждый следующий нарастающий фронт на входе CD остается без последствий.

Уровень сигнала 1 на входе LOAD загружает в счетчик предварительно установленное значение независимо от значения на входе CD.

На выходе Q отображается, является ли текущее счетное значение меньшим или равным нулю.

SFB1 "CTD" соответствует стандарту IEC 1131-3.

Операционная система сбрасывает экземпляры SFB1 "CTD" во время холодного рестарта. Если Вы хотите, чтобы экземпляры этого SFB были инициализированы после теплого рестарта, Вы должны вызвать SFB1 с  $LOAD = 1$  и  $PV =$  требуемому начальному значению для CV в OB100. Если экземпляры этого SFB расположены внутри другого блока, то Вы можете сбросить эти экземпляры, например, инициализацией блока более высокого уровня.

Параметр	Описание	Тип данных	Область памяти	Характеристика
CD	INPUT	BOOL	I, Q, M, D, L, константа	Счетный вход
LOAD	INPUT	BOOL	I, Q, M, D, L, константа	Вход загрузки. LOAD имеет преимущество над CD.
PV	INPUT	INT	I, Q, M, D, L, константа	Предварительно заданное значение. Счетчик устанавливается на PV, когда уровень сигнала на входе LOAD равен 1.
Q	OUTPUT	BOOL	I, Q, M, D, L	Состояние счетчика: Q имеет значение: <ul style="list-style-type: none"> <li>• 1, если <math>CV \leq 0</math></li> <li>• 0 в противном случае</li> </ul>
CV	OUTPUT	INT	I, Q, M, D, L	Текущее значение счетчика (возможные значения: от -32 768 до 32 767)

## 23.6 Прямой и обратный счет с помощью SFB2 "CTUD"

### Описание

С помощью SFB 2 "CTUD" Вы можете вести прямой и обратный счет. Значение счетчика при нарастающем фронте изменяется по сравнению с последним вызовом SFB следующим образом:

- на CU он увеличивается на 1
- на CD он уменьшается на 1.

Если счетное значение достигает границ, то счетчик реагирует следующим образом:

- нижняя граница -32768, он больше не уменьшается
- верхняя граница 32767, он больше не увеличивается.

Если в одном цикле имеет место нарастающий фронт как на входе CU, так и на входе CD, то счетчик сохраняет свое текущее значение. Эта реакция не соответствует стандарту IEC 1131-3. В соответствии со стандартом вход CU имеет преимущество, если оба сигнала активны одновременно. Это изменение предложено IEC (International Electrotechnical Commission, Международной электротехнической комиссии).

Уровень сигнала 1 на входе LOAD предварительно устанавливает счетчик на значение PV независимо от значений на входах CU и CD.

Уровень сигнала 1 на входе R сбрасывает счетчик на значение 0 независимо от значений на входах CU, CD и LOAD. Выход QU показывает, является ли текущее счетное значение большим или равным предварительно заданному значению PV; выход QD показывает, является ли это значение меньшим или равным 0.

Операционная система сбрасывает SFB2 "CTUD" во время холодного рестарта. Если Вы хотите, чтобы SFB2 "CTUD" инициализировался после теплового рестарта, Вы должны вызвать SFB2 в OB100 следующим образом:

- с  $R = 1$  при использовании блока для прямого счета
- с  $R = 0$  и  $LOAD = 1$  и  $PV =$  требуемому начальному значению для CV при использовании блока для обратного счета

Если экземпляры этого SFB расположены внутри другого блока, то Вы можете сбросить эти экземпляры, например, инициализацией блока более высокого уровня.

Параметр	Описание	Тип данных	Область памяти	Характеристика
CU	INPUT	BOOL	I, Q, M, D, L, константа	Вход прямого счета.
CD	INPUT	BOOL	I, Q, M, D, L, константа	Вход обратного счета
R	INPUT	BOOL	I, Q, M, D, L, константа	Вход сброса. R имеет преимущество над LOAD.
LOAD	INPUT	BOOL	I, Q, M, D, L, константа	Вход загрузки. LOAD имеет преимущество над CU и CD.
PV	INPUT	INT	I, Q, M, D, L, константа	Предварительно заданное значение. Счетчик устанавливается на предварительно заданное значение PV, когда уровень сигнала на входе LOAD равен 1.
QU	OUTPUT	BOOL	I, Q, M, D, L	Состояние прямого счетчика: QU имеет следующее значение <ul style="list-style-type: none"> <li>• 1, если <math>CV \geq PV</math></li> <li>• 0 в противном случае</li> </ul>
QD	OUTPUT	BOOL	I, Q, M, D, L	Состояние обратного счетчика: QD имеет следующее значение <ul style="list-style-type: none"> <li>• 1, если <math>CV \leq 0</math></li> <li>• 0 в противном случае</li> </ul>
CV	OUTPUT	INT	I, Q, M, D, L	Текущее значение счетчика (возможные значения: от -32 768 до 32 767)



## 24 Функции IEC

## 24.1 Обзор

### Имеющиеся в распоряжении функции

Вы можете копировать следующие функции Международной электротехнической комиссии (International Electrotechnical Commission, IEC) из библиотеки S7libs\Stdlib30 пакета STEP 7 в свою папку с программами.

Имя	Семейство блоков IEC	Функция
D_TOD_DT	FC 3	Convert [Преобразование] Объединение DATE и TIME_OF_DAY в DT
DT_DATE	FC 6	Convert Извлечение DATE из DT
DT_DAY	FC 7	Convert Извлечение дня недели из DT
DT_TOD	FC 8	Convert Извлечение TIME_OF_DAY из DT
S5TI_TIM	FC 33	Convert Преобразование типа данных S5TIME в TIME
TIM_S5TI	FC 40	Convert Преобразование типа данных TIME в S5TIME
I_STRNG	FC 16	Convert Преобразование типа данных INT в STRING
DI_STRNG	FC 5	Convert Преобразование типа данных DINT в STRING
R_STRNG	FC 30	Convert Преобразование типа данных REAL в STRING
STRNG_I	FC 38	Convert Преобразование типа данных STRING в INT
STRNG_DI	FC 37	Convert Преобразование типа данных STRING в DINT
STRNG_R	FC 39	Convert Преобразование типа данных STRING в REAL
EQ_DT	FC 9	DT Сравнение DT на равно
GE_DT	FC 12	DT Сравнение DT на больше или равно
GT_DT	FC 14	DT Сравнение DT на больше
LE_DT	FC 18	DT Сравнение DT на меньше или равно
LT_DT	FC 23	DT Сравнение DT на меньше
NE_DT	FC 28	DT Сравнение DT на не равно
EQ_STRNG	FC 10	String [строка] Сравнение STRING на равно
GE_STRNG	FC 13	String Сравнение STRING на больше или равно
GT_STRNG	FC 15	String Сравнение STRING на больше
LE_STRNG	FC 19	String Сравнение STRING на меньше или равно
LT_STRNG	FC 24	String Сравнение STRING на меньше
NE_STRNG	FC 29	String Сравнение STRING на не равно
LEN	FC 21	String Длина переменной STRING
LEFT	FC 20	String Левая часть переменной STRING
RIGHT	FC 32	String Правая часть переменной STRING
MID	FC 26	String Средняя часть переменной STRING
CONCAT	FC 2	String Объединение двух переменных STRING
INSERT	FC 17	String Вставка в переменной STRING
DELETE	FC 4	String Удаление в переменной STRING
REPLACE	FC 31	String Замена в переменной STRING
FIND	FC 11	String Поиск в переменной STRING
AD_DT_TM	FC 1	Floating-Point Math [математика с плавающей точкой] Прибавление промежутка времени к моменту времени
SB_DT_TM	FC 35	Floating-Point Math Вычитание промежутка времени из момента времени
SB_DT_DT	FC 34	Floating-Point Math Вычитание двух значений времени
LIMIT	FC 22	Floating-Point Math Ограничение
MAX	FC 25	Floating-Point Math Выбор максимума
MIN	FC 27	Floating-Point Math Выбор минимума
SEL	FC 36	Floating-Point Math Двоичный выбор

## 24.2 Технические данные функций IEC

### Требования к памяти

Следующая таблица показывает, сколько требуется рабочей и загружаемой памяти для каждой функции IEC, а также количество байтов локальных данных, требуемых для каждой функции IEC.

Имя		Байты рабочей памяти	Байты загрузочной памяти	Байты локальных данных
D_TOD_DT	FC3	634	810	12
DT_DATE	FC6	340	466	10
DT_DAY	FC7	346	472	10
DT_TOD	FC8	114	210	6
S5TI_TIM	FC33	94	208	2
TIM_S5TI	FC40	104	208	6
I_STRNG	FC16	226	340	10
DI_STRNG	FC5	314	440	18
R_STRNG	FC30	528	684	28
STRNG_I	FC38	292	420	12
STRNG_DI	FC37	310	442	12
STRNG_R	FC39	828	1038	30
EQ_DT	FC9	96	194	2
GE_DT	FC12	174	288	4
GT_DT	FC14	192	310	4
LE_DT	FC18	168	280	4
LT_DT	FC23	192	310	4
NE_DT	FC28	96	194	2
EQ_STRNG	FC10	114	220	4
GE_STRNG	FC13	162	282	8
GT_STRNG	FC15	158	278	8
LE_STRNG	FC19	162	282	8
LT_STRNG	FC24	158	278	8
NE_STRNG	FC29	150	266	8
LEN	FC21	38	132	2
LEFT	FC20	200	320	8
RIGHT	FC32	230	350	8
MID	FC26	302	390	8
CONCAT	FC2	358	452	14
INSERT	FC17	488	644	20
DELETE	FC4	376	512	8
REPLACE	FC31	562	726	20
FIND	FC11	236	360	14
AD_DT_TM	FC1	1350	1590	22
SB_DT_TM	FC35	1356	1596	22
SB_DT_DT	FC34	992	1178	30
LIMIT	FC22	426	600	12
MAX	FC25	374	532	8
MIN	FC27	374	532	8
SEL	FC36	374	560	8

## 24.3 Дата и время как составные типы данных

### Фактические параметры для DATE\_AND\_TIME

Тип данных DATE\_AND\_TIME относится к категории составных типов данных, наряду с ARRAY, STRING и STRUCT. Допустимыми областями памяти для составных типов данных являются блоки данных (DB) и области локальных данных (L-стек).

Так как DATE\_AND\_TIME – это составной тип данных, то когда Вы используете его в качестве формального параметра в команде, Вы должны определять фактический параметр в одной из следующих форм:

- как локальный символ блока из таблицы описания переменных конкретного блока
- как символическое имя для блока данных, например, "DB\_sys\_info.System\_Time", составленное из следующих двух частей:
  - имя, определенное в таблице символов для номера блока данных (например, "DB\_sys\_info" для DB5)
  - имя, определенное внутри блока данных для элемента DATE\_AND\_TIME (например, "System\_Time" для переменной типа данных DATE\_AND\_TIME, которая содержится в DB5)

Вы не можете использовать константы в качестве фактических параметров для формальных параметров, принадлежащих к составным типам данных, включая DATE\_AND\_TIME. Вы также не можете передавать абсолютные адреса в качестве фактических параметров для DATE\_AND\_TIME.



## 24.4 Функции даты и времени суток

### Описание FC1 AD\_DT\_TM

Функция FC 1 прибавляет промежуток времени (формат TIME) к моменту времени (формат DT) и поставляет в качестве результата новый момент времени (формат DT). Момент времени (параметр T) должен лежать в диапазоне от DT#1990-01-01-00:00:00.000 до DT#2089-12-31-23:59:59.999. Функция не выполняет входной проверки. Если результат сложения не лежит внутри допустимого диапазона, то результат ограничивается соответствующим значением и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
T	INPUT	DT	I, Q, M, D, L	Момент времени в формате DT
D	INPUT	TIME	I, Q, M, D, L, константа	Промежуток времени в формате TIME
RET_VAL	OUTPUT	DT	I, Q, M, D, L	Сумма в формате DT

Для входного параметра T и выходного параметра можно ставить в соответствие только символически определенную переменную.

### Описание FC3 D\_TOD\_DT

Функция FC 3 объединяет форматы данных DATE и TIME\_OF\_DAY (TOD) и преобразует эти форматы в формат данных DATE\_AND\_TIME (DT). Входное значение IN1 должно лежать между границами DATE#1990-01-01 и DATE#2089-12-31 (это значение не проверяется). Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	DATE	I, Q, M, D, L, константа	Входная переменная в формате DATE
IN2	INPUT	TIME_OF_DAY	I, Q, M, D, L, константа	Входная переменная в формате TOD
RET_VAL	OUTPUT	DATE_AND_TIME	D, L	Возвращаемое значение в формате DT

Возвращаемому значению можно ставить в соответствие только символически определенную переменную.

**Описание FC6 DT\_DATE**

Функция FC6 извлекает формат данных DATE из формата данных DATE\_AND\_TIME. Значение DATE должно находиться между граничными значениями DATE#1990-1-1 и DATE#2089-12-31. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	DATE	I, Q, M, D, L	Возвращаемое значение в формате DATE

Входному значению можно ставить в соответствие только символически определенную переменную.

**Описание FC7 DT\_DAY**

Функция IEC FC7 извлекает день недели из формата DATE\_AND\_TIME. недели представляется в формате данных INT (от 1 до 7):

- 1 Воскресенье
- 2 Понедельник
- 3 Вторник
- 4 Среда
- 5 Четверг
- 6 Пятница
- 7 Суббота

Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Возвращаемое значение в формате INT

Входному значению можно ставить в соответствие только символически определенную переменную.

**Описание FC8 DT\_TOD**

Функция IEC FC8 извлекает формат данных TIME\_OF\_DAY из формата DATE\_AND\_TIME. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	TIME_OF_DAY	I, Q, M, D, L	Возвращаемое значение в формате TOD

Входному значению можно ставить в соответствие только символически определенную переменную.

**Описание FC33 S5TI\_TIM**

Функция IEC FC33 преобразует формат данных S5TIME в формат TIME. Если результат преобразования лежит вне диапазона TIME, то результат ограничивается соответствующим значением и бит двоичного результата (BR) слова состояний устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	S5TIME	I, Q, M, D, L, константа	Входная переменная в формате S5TIME
RET_VAL	OUTPUT	TIME	I, Q, M, D, L	Возвращаемое значение в формате TIME

**Описание FC34 SB\_DT\_DT**

Функция FC34 вычитает два значения времени (формат DT) иставляет в качестве результата интервал времени (формат TIME). Значения времени должны лежать в диапазоне от DT#1990-01-01-00:00:00.000 до DT#2089-12-31-23:59:59.999. Функция не выполняет входной проверки. Если первое время (параметр T1) больше (младше), чем второе (параметр T2), то результат положителен; если первое время меньше (старше), чем второе, то результат отрицателен. Если результат вычитания лежит вне диапазона TIME, то результат ограничивается соответствующим значением и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DT	I, Q, M, D, L	Первый момент времени в формате DT
DT2	INPUT	DT	I, Q, M, D, L	Второй момент времени в формате DT
RET_VAL	OUTPUT	TIME	I, Q, M, D, L	Разность в формате TIME

Входным параметрам Вы можете ставить в соответствие только символически определенную переменную.

**Описание FC35 SB\_DT\_TM**

Функция FC 35 вычитает промежуток времени (формат TIME) из момента времени (формат DT) иставляет в качестве результата новый момент времени (формат DT). Момент времени (параметр T) должен лежать в диапазоне от DT#1990-01-01-00:00:00.000 до DT#2089-12-31-23:59:59.999. Функция не выполняет входной проверки. Если результат вычитания не лежит внутри допустимого диапазона, то результат ограничивается соответствующим значением и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
T	INPUT	DT	I, Q, M, D, L	Момент времени в формате DT
D	INPUT	TIME	I, Q, M, D, L, константа	Промежуток времени в формате TIME
RET_VAL	OUTPUT	DT	I, Q, M, D, L	Разность в формате DT

Для входного параметра T и выходного параметра можно ставить в соответствие только символически определенную переменную

### Описание FC40 TIM\_S5TI

Функция FC 40 преобразует формат данных TIME в формат S5TIME. При преобразовании происходит округление. Если входной параметр больше, чем допускает формат S5TIME (больше, чем TIME#02:46:30.000), то выводится результат S5TIME#999.3 и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	TIME	I, Q, M, D, L, константа	Входная переменная в формате TIME
RET_VAL	OUTPUT	S5TIME	I, Q, M, D, L	Возвращаемое значение в формате S5TIME

## 24.5 Сравнение переменных типа DATE\_AND\_TIME

### Описание FC9 EQ\_DT

Функция FC 9 сравнивает содержимое двух переменных в формате DATE\_AND\_TIME, чтобы выяснить, равны ли они, и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала "1", если момент времени в параметре DT1 равен моменту времени в параметре DT2. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
DT2	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC12 GE\_DT

Функция FC 12 сравнивает содержимое двух переменных в формате DATE\_AND\_TIME на "больше или равно" и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала "1", если момент времени в параметре DT1 больше (младше), чем момент времени в параметре DT2, или оба момента времени равны. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
DT2	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC14 GT\_DT

Функция FC14 сравнивает содержимое двух переменных в формате DATE\_AND\_TIME на "больше" и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала "1", если момент времени в параметре DT1 больше (младше), чем момент времени в параметре DT2. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
DT2	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

**Описание FC18 LE\_DT**

Функция FC 18 сравнивает содержимое двух переменных в формате DATE\_AND\_TIME на “меньше или равно” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если момент времени в параметре DT1 меньше (старше), чем момент времени в параметре DT2, или оба момента времени равны. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
DT2	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

**Описание FC23 LT\_DT**

Функция FC 23 сравнивает содержимое двух переменных в формате DATE\_AND\_TIME на “меньше” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если момент времени в параметре DT1 меньше (старше), чем момент времени в параметре DT2. Функция не сообщает об ошибке.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
DT2	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

**Описание FC28 NE\_DT**

Функция FC 28 сравнивает значения двух переменных в формате DATE\_AND\_TIME на “не равно” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если момент времени в параметре DT1 не равен моменту времени в параметре DT2. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
DT1	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
DT2	INPUT	DATE_AND_TIME	D, L	Входная переменная в формате DT
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

## 24.6 Сравнение переменных типа STRING

### Описание FC10 EQ\_STRNG

Функция FC 10 сравнивает значения двух переменных в формате STRING на “равно” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 равна строке символов в параметре S2. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S1	INPUT	STRING	D, L	Входная переменная в формате STRING
S2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC13 GE\_STRNG

Функция FC 13 сравнивает содержимое двух переменных в формате STRING на “больше или равно” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 больше, чем строка символов в параметре S2, или равна ей. Начиная слева, символы сравниваются по их ASCII-кодам (например, ‘a’ больше, чем ‘A’). Первый отличный символ определяет результат сравнения. Если левая часть более длинной строки символов идентична более короткой строке символов, то более длинная строка символов считается большей. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S1	INPUT	STRING	D, L	Входная переменная в формате STRING
S2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC15 GT\_STRNG

Функция FC 15 сравнивает содержимое двух переменных в формате STRING на “больше” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 больше, чем строка символов в параметре S2. Начиная слева, символы сравниваются по их ASCII-кодам (например, ‘a’ больше, чем ‘A’). Первый отличный символ определяет результат сравнения. Если левая часть более длинной строки символов идентична более короткой строке символов, то более длинная строка символов считается большей. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S1	INPUT	STRING	D, L	Входная переменная в формате STRING
S2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC19 LE\_STRNG

Функция FC 19 сравнивает содержимое двух переменных в формате STRING на “меньше или равно” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 меньше, чем строка символов в параметре S2, или равна ей. Начиная слева, символы сравниваются по их ASCII-кодам (например, ‘A’ меньше, чем ‘a’). Первый отличный символ определяет результат сравнения. Если левая часть более длинной строки символов идентична более короткой строке символов, то более короткая строка символов считается меньшей. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S1	INPUT	STRING	D, L	Входная переменная в формате STRING
S2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.



### Описание FC24 LT\_STRNG

Функция FC 24 сравнивает содержимое двух переменных в формате STRING на “меньше” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 меньше, чем строка символов в параметре S2. Начиная слева, символы сравниваются по их ASCII-кодам (например, ‘A’ меньше, чем ‘a’). Первый отличный символ определяет результат сравнения. Функция FC 24 сравнивает содержимое двух переменных в формате STRING на “меньше” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 меньше, чем строка символов в параметре S2. Начиная слева, символы сравниваются по их ASCII-кодам (например, ‘A’ меньше, чем ‘a’). Первый отличный символ определяет результат сравнения. Если левая часть более длинной строки символов идентична более короткой строке символов, то более короткая строка символов считается меньшей. Функция не сообщает об ошибке.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S1	INPUT	STRING	D, L	Входная переменная в формате STRING
S2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC29 NE\_STRNG

Функция FC 29 сравнивает содержимое двух переменных в формате STRING на “не равно” и выводит результат сравнения в качестве возвращаемого значения. Возвращаемое значение имеет состояние сигнала “1”, если строка символов в параметре S1 не равна строке символов в параметре S2. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S1	INPUT	STRING	D, L	Входная переменная в формате STRING
S2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	BOOL	I, Q, M, D, L	Результат сравнения

Входным параметрам можно ставить в соответствие только символически определенную переменную.

## 24.7 Редактирование числовых значений

### Описание FC22 LIMIT

Функция FC 22 ограничивает числовое значение переменной выбираемыми граничными значениями. В качестве входных значений допускаются переменные типов INT, DINT и REAL. Все переменные с назначенными параметрами должны иметь тот же тип данных. Нижнее граничное значение (параметр MN) не должен быть больше, чем верхнее граничное значение (параметр MX).

Выходное значение остается неизменным и бит двоичного результата (BR) слова состояния устанавливается в "0", если выполняется любое из следующих условий:

- переменная с назначенными параметрами имеет недопустимый тип данных
- не все переменные с назначенными параметрами имеют один и тот же тип данных
- нижнее граничное значение больше, чем верхнее граничное значение
- переменная типа REAL не представляет допустимое число с плавающей точкой.

Параметр	Описание	Тип данных	Область памяти	Характеристика
MN	INPUT	ANY	I, Q, M, D, L	Нижняя граница
IN	INPUT	ANY	I, Q, M, D, L	Входная переменная
MX	INPUT	ANY	I, Q, M, D, L	Верхняя граница
RET_VAL	OUTPUT	ANY	I, Q, M, D, L	Ограниченная выходная переменная

### Описание FC25 MAX

Функция FC 25 выбирает наибольшее из трех числовых значений переменных. В качестве входных значений допускаются значения переменных, принадлежащих к типам данных INT, DINT и REAL. Все переменные с назначенными параметрами должны принадлежать к одному типу данных.

Выходное значение остается неизменным и бит двоичного результата (BR) слова состояния устанавливается в "0", если выполняется любое из следующих условий:

- переменная с назначенными параметрами имеет недопустимый тип данных
- не все переменные с назначенными параметрами имеют один и тот же тип данных
- переменная типа REAL не представляет допустимое число с плавающей точкой.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	ANY	I, Q, M, D, L	Первое входное значение
IN2	INPUT	ANY	I, Q, M, D, L	Второе входное значение
IN3	INPUT	ANY	I, Q, M, D, L	Третье входное значение
RET_VAL	OUTPUT	ANY	I, Q, M, D, L	Наибольшее из входных значений

## 24.8 Пример на STL

```

CALL FC 25
  IN1          := P#M 10.0 DINT 1
  IN2          := MD20
  IN3          := P#DB1.DBX 0.0 DINT 1
  RET_VAL     := P#M 40.0 DINT 1
=             M 0.0

```

Примечание:

Допустимые типы данных INT, DINT и REAL должны быть введены в указатель ANY.

Такие параметры, как "MD20" также допустимы, однако, Вы должны определить соответствующий тип данных "MD20" в "Symbol".

### Описание FC27 MIN

Функция FC 27 выбирает наименьшее из трех числовых значений переменных. В качестве входных значений допускаются значения переменных, принадлежащих к типам данных INT, DINT и REAL. Все переменные с назначенными параметрами должны принадлежать к одному типу данных.

Выходное значение остается неизменным и бит двоичного результата (BR) слова состояния устанавливается в "0", если выполняется любое из следующих условий:

- переменная с назначенными параметрами имеет недопустимый тип данных
- не все переменные с назначенными параметрами имеют один и тот же тип данных
- переменная типа REAL не представляет допустимое число с плавающей точкой.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	ANY	I, Q, M, D, L	Первое входное значение
IN2	INPUT	ANY	I, Q, M, D, L	Второе входное значение
IN3	INPUT	ANY	I, Q, M, D, L	Третье входное значение
RET_VAL	OUTPUT	ANY	I, Q, M, D, L	Наименьшее из входных значений

## 24.9 Пример на STL

```

CALL FC 27
  IN1           := P#M 10.0 DINT 1
  IN2           := MD20
  IN3           := P#DB1.DBX 0.0 DINT 1
  RET_VAL       := P#M 40.0 DINT 1
=              M 0.0

```

Примечание:

Допустимые типы данных INT, DINT и REAL должны быть введены в указатель ANY.

Такие параметры, как "MD20" также допустимы, однако, Вы должны определить соответствующий тип данных "MD20" в "Symbol".

### Описание FC36 SEL

Функция FC 36 выбирает одно из двух числовых значений переменных в зависимости от переключателя (параметр G). В качестве входных значений IN0 и IN1 допускаются значения переменных, принадлежащих к типам данных с размером бит, байт, слово и двойное слово (кроме типов DT и STRING). Обе входные и выходная переменные должны принадлежать к одному типу данных.

Выходное значение остается неизменным и бит двоичного результата (BR) слова состояния устанавливается в "0", если выполняется любое из следующих условий:

- переменная с назначенными параметрами имеет недопустимый тип данных
- не все переменные с назначенными параметрами имеют один и тот же тип данных
- переменная типа REAL не представляет допустимое число с плавающей точкой.

Параметр	Описание	Тип данных	Область памяти	Характеристика
G	INPUT	BOOL	I, Q, M, D, L	Переключающий параметр
IN0	INPUT	ANY	I, Q, M, D, L	Первое входное значение
IN1	INPUT	ANY	I, Q, M, D, L	Второе входное значение
RET_VAL	OUTPUT	ANY	I, Q, M, D, L	Выбранное из двух входных значений

## 24.10 Редактирование переменных типа STRING

### Описание FC2 CONCAT

Функция FC 2 объединяет две переменные типа STRING в одну строку символов. Если результирующая строка символов длиннее, чем переменная, заданная в выходном параметре, то результирующая строка ограничивается максимально установленной длиной и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	STRING	D, L	Входная переменная в формате STRING
IN2	INPUT	STRING	D, L	Входная переменная в формате STRING
RET_VAL	OUTPUT	STRING	D, L	Объединенная строка

Параметрам можно ставить в соответствие только символически определенную переменную.

### Описание FC4 DELETE

Функция FC 4 удаляет L символов в строке, начиная с P-го символа (включительно). Если L и/или P равны нулю или P больше, чем текущая длина входной строки, то возвращается входная строка. Если сумма L и P больше, чем входная строка символов, то строка символов удаляется до конца. Если L и/или P отрицательны, то выводится пустая строка и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	STRING	D, L	Переменная типа STRING, в которой происходит удаление
L	INPUT	INT	I, Q, M, D, L, константа	Количество удаляемых символов
P	INPUT	INT	I, Q, M, D, L, константа	Позиция 1-го удаляемого символа
RET_VAL	OUTPUT	STRING	D, L	Результирующая строка

Входному параметру IN и выходному параметру можно ставить в соответствие только символически определенную переменную.

### Описание FC11 FIND

Функция FC 11 сообщает позицию второй строки символов (IN2) внутри первой строки символов (IN1). Поиск начинается слева; сообщается о первом появлении строки символов. Если вторая строка символов не содержится в первой, то возвращается нуль. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	STRING	D, L	Переменная типа STRING, в которой происходит поиск
IN2	INPUT	STRING	D, L	Искомая переменная типа STRING
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Позиция найденной строки символов

Входным параметрам IN1 и IN2 можно ставить в соответствие только символически определенную переменную.

### Описание FC17 INSERT

Функция FC 17 вставляет строку символов из параметра IN2 в строку символов в параметре IN1 после P-го символа.

Если P равно нулю, то вторая строка символов вставляется перед первой строкой символов. Если P больше, чем текущая длина первой строки символов, то вторая строка символов присоединяется к первой.

Если P отрицательно, то выводится пустая строка и бит двоичного результата (BR) слова состояния устанавливается в "0". Бит двоичного результата также устанавливается в "0", если 0" результирующая строка длиннее, чем переменная, заданная в выходном параметре; в этом случае результирующая строка ограничивается максимально установленной длиной.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	STRING	D, L	Переменная типа STRING, в которую происходит вставка
IN2	INPUT	STRING	D, L	Вставляемая переменная типа STRING
P	INPUT	INT	I, Q, M, D, L, константа	Позиция вставки
RET_VAL	OUTPUT	STRING	D, L	Результирующая строка

Входным параметрам IN1 и IN2 и выходному параметру можно ставить в соответствие только символически определенную переменную.

### Описание FC20 LEFT

Функция FC 20 выводит первые L символов строки. Если L больше, чем текущая длина переменной типа STRING, то возвращается входное значение. При L = 0 и при пустой строке в качестве входного значения возвращается пустая строка. Если число L отрицательно, то выводится пустая строка и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	STRING	D, L	Входная переменная в формате STRING
L	INPUT	INT	I, Q, M, D, L, константа	Длина левой части строки
RET_VAL	OUTPUT	STRING	D, L	Выходная переменная в формате STRING

Параметру IN и возвращаемому значению можно ставить в соответствие только символически определенную переменную.

### Описание FC21 LEN

Переменная типа STRING содержит две длины: максимальную длину (она задается при определении переменных в квадратных скобках) и текущую длину (это количество символов, действующих в данный момент времени). Текущая длина должна быть меньше, чем максимальная длина, или равна ей. Количество байтов, занимаемых строкой символов, больше максимальной длины на 2.

Функция FC 21 выводит текущую длину строки (количество действующих символов) в качестве возвращаемого значения. Пустая строка (") имеет длину нуль. Максимальная длина равна 254. Функция не сообщает об ошибках.

Параметр	Описание	Тип данных	Область памяти	Характеристика
S	INPUT	STRING	D, L	Переменная типа STRING
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Количество действующих символов

Входному параметру можно ставить в соответствие только символически определенную переменную.

### Описание FC26 MID

Функция FC 26 выводит среднюю часть строки символов (L символов, начиная с P-го символа включительно). Если сумма L и (P-1) превосходит текущую длину переменной типа STRING, то возвращается строка символов, начиная с P-го символа входной строки до ее конца. Во всех остальных случаях (P находится вне текущей длины, P и/или L равны нулю или отрицательны) выводится пустая строка и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	STRING	D, L	Входная переменная в формате STRING
L	INPUT	INT	I, Q, M, D, L, константа	Длина средней части строки
P	INPUT	INT	I, Q, M, D, L, константа	Позиция первого символа
RET_VAL	OUTPUT	STRING	D, L	Выходная переменная в формате STRING

Параметру IN и возвращаемому значению можно ставить в соответствие только символически определенную переменную.

### Описание FC31 REPLACE

Функция FC 31 заменяет L символов первой строки (IN1), начиная с P-го символа (включительно) второй строкой символов (IN2). Если L равно нулю, то возвращается первая строка символов. Если P равно нулю или единице, то замена происходит, начиная с 1-го символа (включительно). Если P лежит вне первой строки символов, то вторая строка присоединяется к первой строке. Если L и/или P отрицательны, то возвращается пустая строка и бит двоичного результата (BR) слова состояния устанавливается в "0". Бит двоичного результата также устанавливается в "0", результирующая строка длиннее, чем переменная, заданная в выходном параметре; в этом случае результирующая строка ограничивается максимально установленной длиной.

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN1	INPUT	STRING	D, L	Переменная типа STRING, в которой происходит замена
IN2	INPUT	STRING	D, L	Заменяющая переменная типа STRING
L	INPUT	INT	I, Q, M, D, L, константа	Количество заменяемых символов
P	INPUT	INT	I, Q, M, D, L, константа	Позиция 1-го заменяемого символа
RET_VAL	OUTPUT	STRING	D, L	Результирующая строка

Входным параметрам IN1 и IN2 и выходному параметру можно ставить в соответствие только символически определенную переменную.



### Описание FC32 RIGHT

Функция FC 32 выводит последние L символов строки (где L означает число). Если L больше, чем текущая длина переменной STRING, то возвращается входное значение. При L = 0 и при пустой строке в качестве входного значения возвращается пустая строка. Если число L отрицательно, то выводится пустая строка и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
IN	INPUT	STRING	D, L	Входная переменная в формате STRING
L	INPUT	INT	I, Q, M, D, L, константа	Длина правой части строки
RET_VAL	OUTPUT	STRING	D, L	Выходная переменная в формате STRING

Параметру IN и возвращаемому значению можно ставить в соответствие только символически определенную переменную.

## 24.11 Преобразование форматов типов данных

### Описание FC5 DI\_STRNG

Функция FC 5 преобразует переменную в формате DINT в строку символов. Строка символов представляется с предшествующим знаком. Если переменная, заданная для возвращаемого параметра, слишком коротка, то преобразование не происходит и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
I	INPUT	DINT	I, Q, M, D, L, константа	Входное значение
RET_VAL	OUTPUT	STRING	D, L	Результирующая строка

Для выходного параметра можно ставить в соответствие только символически определенную переменную.

### Описание FC16 I\_STRNG

Функция FC 16 преобразует переменную в формате INT в строку символов. Строка символов представляется с предшествующим знаком. Если переменная, заданная для возвращаемого параметра, слишком коротка, то преобразование не происходит и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
I	INPUT	INT	I, Q, M, D, L, константа	Входное значение
RET_VAL	OUTPUT	STRING	D, L	Результирующая строка

Для выходного параметра можно ставить в соответствие только символически определенную переменную.

### Описание FC30 R\_STRNG

Функция FC 30 преобразует переменную в формате REAL в строку символов. Строка символов представляется 14 разрядами:

$\pm v.nnnnnnnE\pm xx$

- $\pm$  знак
- v 1 разряд перед десятичной точкой
- n 7 разрядов после десятичной точки
- x 2 разряда показателя степени

Если заданная для возвращаемого параметра переменная слишком коротка или если параметру IN задано недопустимое число с плавающей точкой, то преобразование не происходит и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
I	INPUT	REAL	I, Q, M, D, L, константа	Входное значение
RET_VAL	OUTPUT	STRING	D, L	Результирующая строка

Выходному параметру можно ставить в соответствие только символически определенную переменную.

### Описание FC37 STRNG\_DI

Функция FC 37 преобразует строку символов в переменную в формате DINT. Первый символ в строке символов может быть знаком или цифрой, следующие после этого символы должны состоять из цифр. Если длина строки символов равна нулю или больше 11 или в строке символов находятся неразрешенные символы, то преобразование не происходит и бит двоичного результата (BR) слова состояния устанавливается в "0". Если результат преобразования лежит вне области DINT, то результат ограничивается соответствующим значением и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
S	INPUT	STRING	D, L	Входная строка
RET_VAL	OUTPUT	DINT	I, Q, M, D, L	Результат

Входному параметру можно ставить в соответствие только символически определенную переменную.

### Описание FC38 STRNG\_I

Функция FC 38 преобразует строку символов в переменную в формате INT. Первый символ в строке символов может быть знаком или цифрой, следующие после этого символы должны состоять из цифр. Если длина строки символов равна нулю или больше 6 или в строке символов находятся неразрешенные символы, то преобразование не происходит и бит двоичного результата (BR) слова состояния устанавливается в "0". Если результат преобразования лежит вне области INT, то результат ограничивается соответствующим значением и бит двоичного результата (BR) слова состояния устанавливается в "0".

Параметр	Описание	Тип данных	Область памяти	Характеристика
S	INPUT	STRING	D, L	Входная строка
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Результат

Входному параметру можно ставить в соответствие только символически определенную переменную.



## **25 SFB с интегрированными функциями регулирования**

## 25.1 Непрерывное регулирование с помощью SFB 41 /FB 41 "CONT\_C"

### Введение

SFB "CONT\_C" используется в программируемых логических контроллерах SIMATIC S7 для управления техническими процессами с непрерывными входными и выходными переменными. При назначении параметров Вы можете активировать и деактивировать отдельные функции ПИД-регулятора, чтобы адаптировать его к процессу. Вы можете легко выполнить назначения, используя утилиту для назначения параметров (из меню: **Start** [Пуск] > **Simatic** > **Step7** > **Assign PID Control parameters** [Назначение параметров ПИД-регулирования]). Интерактивное электронное справочное руководство можно найти в меню: **Start** [Пуск] > **Simatic** > **Step7** > **Assign PID Control English** [Назначение параметров ПИД-регулирования]).

### Применение

Этот регулятор можно использовать как ПИД-регулятор с постоянными уставками или в многоконтурных системах регулирования в качестве каскадного регулятора, регулятора состава или пропорционального регулятора. Функции регулятора основаны на ПИД-алгоритме регулирования дискретного регулятора с аналоговым сигналом, дополненным, в необходимых случаях, степенью формирования импульсов в целях формирования выходных сигналов с широтно-импульсной модуляцией для двух- или трехпозиционного регулирования с пропорциональными исполнительными звеньями.

### Описание

Наряду с функциями в цепях уставки и цепях переменной процесса, SFB/FB реализует готовый ПИД-регулятор с непрерывным управлением и возможностью ручного воздействия на управляющий сигналы. Ниже Вы найдете подробное описание частных функций:

#### Цепи уставки

Уставка вводится на входе **SP\_INT** в формате с плавающей точкой.

#### Цепи переменной процесса

Переменная процесса может вводиться в формате периферии или в формате с плавающей точкой. Функция **CRP\_IN** преобразует периферийное значение **PV\_PER** в формат с плавающей точкой в диапазоне -100 ... +100 % в соответствии со следующей формулой:

$$\text{Выход CRP\_IN} = \text{PV\_PER} * \frac{100}{27648}$$

Функция **PV\_NORM** нормирует выход **CRP\_IN** в соответствии со следующей формулой:

$$\text{Выход PV\_NORM} = (\text{выход CRP\_IN}) * \text{PV\_FAC} + \text{PV\_OFF}$$

**PV\_FAC** имеет значение по умолчанию, равное 1, а **PV\_OFF** значение по умолчанию, равное 0.

### Сигнал ошибки

Разность между значением уставки и значением переменной процесса называется сигналом ошибки (сигналом рассогласования). Для подавления малых незатухающих колебаний из-за квантованности регулирующего воздействия (например, в случае широтно-импульсной модуляции с помощью PULSEGEN) к сигналу ошибки применяется амплитудный фильтр (DEADBAND). Если DEADB\_W = 0, то амплитудный фильтр выключен.

### ПИД-алгоритм

ПИД-алгоритм работает как алгоритм позиционирования. Пропорциональное, интегрирующая (INT) и дифференцирующая (DIF) составляющие воздействия включены параллельно и могут активироваться и деактивироваться по отдельности. Это дает возможность конфигурировать П-, ПИ-, ПД- и ПИД-регуляторы. Возможны также "чистые" И- и Д-регуляторы.

### Значение, вводимое вручную

Имеется возможность переключения между ручным и автоматическим режимом. В ручном режиме управляющее воздействие корректируется в соответствии со значением, выбранным вручную. Интегратор (INT) внутренне устанавливается на LMN - LMN\_P - DISV, а дифференцирующее устройство (DIF) устанавливается на 0, и производится внутреннее согласование. Это значит, что переключение в автоматический режим не вызывает внезапного изменения управляющего воздействия.

### Управляющее воздействие

Управляющая величина может быть ограничена выбранным значением с помощью функции LMNLIMIT. Пересечение входной величиной границ отображается сигнальными битами.

Функция LMN\_NORM нормирует выход функции LMNLIMIT в соответствии со следующей формулой:

$$LMN = (\text{выход LMNLIMIT}) * LMN\_FAC + LMN\_OFF$$

LMN\_FAC по умолчанию равно 1, а LMN\_OFF по умолчанию равно 0.

Управляющее значение доступно также в периферийном формате. Функция CRP\_OUT преобразует значение с плавающей точкой LMN в периферийное значение в соответствии со следующей формулой:

$$LMN\_PER = LMN * \frac{27648}{100}$$

### Управление с использованием предскажений

Возмущающее воздействие может быть подано на вход **DISV**.

### **Инициализация**

SFB41 "CONT\_C" имеет подпрограмму инициализации, которая прогоняется, когда установлен входной параметр COM\_RST = TRUE.

Во время инициализации интегратор устанавливается на значение инициализации I\_ITVAL. Когда он вызывается в классе приоритета циклических прерываний, он продолжает функционировать, начиная с этого значения.

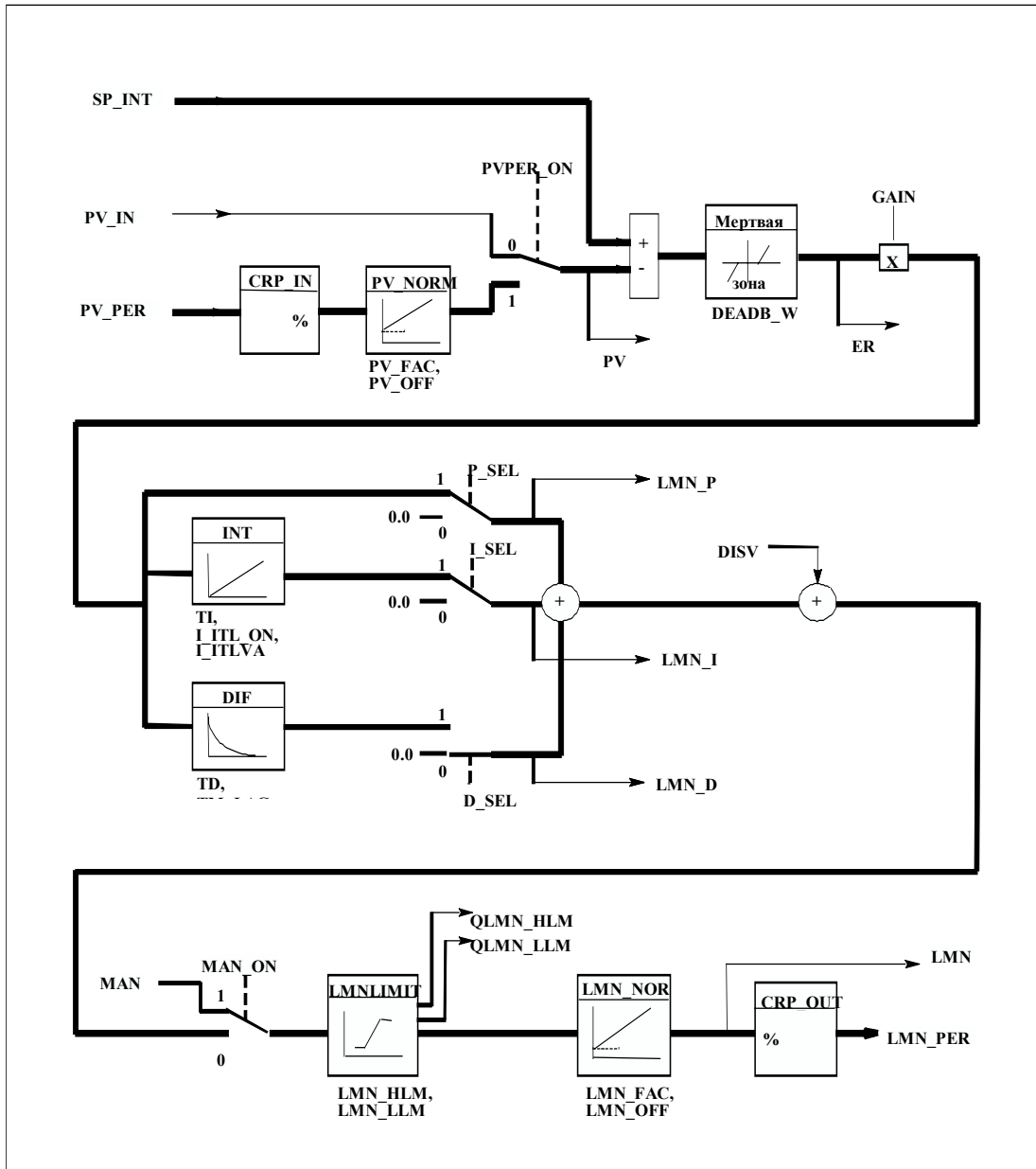
Все остальные выходы устанавливаются на их значения по умолчанию.

### **Информация об ошибках**

Параметр вывода ошибок RET\_VAL не используется.



Блок-схема CONT\_C



**Входные параметры**

Следующая таблица содержит описание входных параметров SFB41 / FB41 "CONT\_C".

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
COM_RST	BOOL		FALSE	COMPLETE RESTART/ Полный рестарт Блок имеет подпрограмму инициализации, которая обрабатывается, когда установлен вход COM_RST ("Полный рестарт").
MAN_ON	BOOL		TRUE	MANUAL VALUE ON / Включение ручного режима Если этот вход установлен, то контур управления разрывается. Значение, заданное вручную, устанавливается в качестве управляющего.
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERAL ON / Включение переменной процесса периферии Если значение переменной процесса считывается из периферии, то вход PV_PER должен быть соединен с периферией, а вход "Включение переменной процесса от периферии" должен быть установлен.
P_SEL	BOOL		TRUE	PROPORTIONAL ACTION ON / Включение пропорционального воздействия В ПИД-алгоритме отдельные воздействия можно активировать или деактивировать индивидуально. Пропорциональное воздействие (П-воздействие) включено, когда установлен вход "Включение пропорционального воздействия".
I_SEL	BOOL		TRUE	INTEGRAL ACTION ON / Включение интегрирующего воздействия В ПИД-алгоритме отдельные воздействия можно активировать или деактивировать индивидуально. Интегрирующее воздействие (И-воздействие) включено, когда установлен вход "Включение интегрирующего воздействия".
INT_HOLD	BOOL		FALSE	INTEGRAL ACTION HOLD / Интегрирующее воздействие заморожено Выход интегратора может быть "заморожен" установкой входа "Интегрирующее воздействие заморожено".
I_ITL_ON	BOOL		FALSE	INITIALIZATION OF THE INTEGRAL ACTION ON / Включение инициализации И-составляющей. Выход интегратора может быть подключен к входу I_ITL_VAL установкой входа "Включение инициализации И-воздействия".
D_SEL	BOOL		FALSE	DERIVATIVE ACTION ON / Включение воздействия дифференциатора. В ПИД-алгоритме отдельные воздействия можно активировать или деактивировать индивидуально. Воздействие дифференциатора (Д-воздействие) включено, когда установлен вход "Включение воздействия Д-компонента".

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
CYCLE	TIME	>= 1 мс	T#1s	SAMPLING TIME / Время опроса Время между вызовами блока должно быть постоянным. Вход "Время опроса" задает время между вызовами блока.
SP_INT	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>1)</sup>	0.0	INTERNAL SETPOINT / Внутреннее значение уставки (внутренняя уставка) Вход "Внутреннее значение уставки" служит для установления заданного значения.
PV_IN	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>1)</sup>	0.0	PROCESS VARIABLE IN / Ввод переменной процесса На входе "Вход переменной процесса" может быть установлено значение инициализации или подключена внешняя переменная процесса в формате с плавающей точкой.
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERAL/ Переменная процесса от периферии Переменная процесса в формате периферии подключается к регулятору на входе " Переменная процесса от периферии".
MAN	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>2)</sup>	0.0	MANUAL VALUE / Ручное значение Вход "Ручное значение" используется для установки ручного значения с помощью функций взаимодействия с оператором.
GAIN	REAL		2.0	PROPORTIONAL GAIN / Коэффициент пропорциональности Вход "Коэффициент пропорциональности" задает коэффициент усиления регулятора.
TI	TIME	>= CYCLE	T#20s	RESET TIME / Время интегрирования Вход "Время интегрирования" определяет временную характеристику интегратора.
TD	TIME	>= CYCLE	T#10s	DERIVATIVE TIME / Время дифференцирования Вход "Время дифференцирования" определяет временную характеристику дифференцирующего звена.
TM_LAG	TIME	>= CYCLE/2	T#2s	TIME LAG OF THE DERIVATIVE ACTION / Время запаздывания Д-воздействия Алгоритм Д-компоненты содержит запаздывание, которое может быть назначено входу "Время запаздывания Д-воздействия".
DEADB_W	REAL	>= 0.0 (%) или физическая величина <sup>1)</sup>	0.0	DEAD BAND WIDTH / Ширина амплитудного фильтра Мертвая зона соответствует ошибке. Вход "Ширина амплитудного фильтра" определяет размер зоны непропускания сигнала.
LMN_HLM	REAL	LMN_LLM ...100.0 (%) или физическая величина <sup>2)</sup>	100.0	MANIPULATED VALUE HIGH LIMIT/ Верхняя граница управляющего воздействия Управляющее воздействие всегда ограничивается верхней и нижней границами. Вход "Верхняя граница управляющего воздействия" задает верхнюю границу.

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
LMN_LLM	REAL	-100.0... LMN_HLM (%) или физическая величина <sup>2)</sup>	0.0	MANIPULATED VALUE LOW LIMIT / Нижняя граница управляющего воздействия Управляющее воздействие всегда ограничивается верхней и нижней границами. Вход "Нижняя граница управляющего воздействия" задает нижнюю границу.
PV_FAC	REAL		1.0	PROCESS VARIABLE FACTOR/ Коэффициент переменной процесса Вход "Коэффициент переменной процесса" умножается на значение переменной процесса. Вход служит для настройки диапазона переменной процесса.
PV_OFF	REAL		0.0	PROCESS VARIABLE OFFSET/ Смещение переменной процесса Вход "Смещение переменной процесса" складывается с фактическим значением. Вход служит для настройки диапазона переменной процесса.
LMN_FAC	REAL		1.0	MANIPULATED VALUE FACTOR Коэффициент управляющего воздействия Вход "Коэффициент управляющего воздействия" умножается на управляющее воздействие. Вход служит для настройки диапазона управляющих воздействий.
LMN_OFF	REAL		0.0	MANIPULATED VALUE OFFSET/ Смещение управляющего воздействия Вход "Смещение управляющего воздействия" складывается с управляющим воздействием. Вход служит для настройки диапазона управляющих воздействий.
I_ITLVAL	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>2)</sup>	0.0	INITIALIZATION VALUE OF THE INTEGRAL ACTION / Инициализирующее значение интегрирующего воздействия Выход интегратора может быть установлен на входе I_ITL_ON. Инициализирующее значение прикладывается к входу "Инициализирующее значение интегрирующего воздействия".
DISV	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>2)</sup>	0.0	DISTURBANCE VARIABLE / Возмущающее воздействие Возмущающее воздействие подключается к входу "Возмущающее воздействие" для упреждающего регулирования.

1) Параметры в цепях уставки и переменной процесса с одинаковой единицей измерения

2) Параметры в цепи управляющей переменной с одинаковой единицей измерения

**Выходные параметры**

Следующая таблица содержит описание выходных параметров SFB41 / FB41 "CONT\_C".

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
LMN	REAL		0.0	MANIPULATED VALUE/ Управляющее воздействие На выходе "Управляющее воздействие" выводится эффективное управляющее воздействие в формате с плавающей точкой.
LMN_PER	WORD		W#16#0000	MANIPULATED VALUE PERIPHERAL / Управляющее воздействие в формате периферии К регулятору на этом выходе подключается управляющее воздействие в формате периферии.
QLMN_HLM	BOOL		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED / Достигнута верхняя граница управляющего воздействия Управляющее воздействие всегда ограничено сверху и снизу. Выход "Достигнута верхняя граница управляющего воздействия" указывает на переход через верхнюю границу.
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED/ Достигнута нижняя граница управляющего воздействия Управляющее воздействие всегда ограничено сверху и снизу. Выход "Достигнута нижняя граница управляющего воздействия" указывает на переход через нижнюю границу.
LMN_P	REAL		0.0	PROPORTIONAL COMPONENT / Пропорциональная составляющая Выход "Пропорциональная составляющая" содержит пропорциональную составляющую управляющего воздействия.
LMN_I	REAL		0.0	INTEGRAL COMPONENT / Интегральная составляющая Выход "Интегральная составляющая" содержит интегральную составляющую управляющего воздействия.
LMN_D	REAL		0.0	DERIVATIVE COMPONENT/ Дифференциальная составляющая Выход "Дифференциальная составляющая" содержит дифференциальную составляющую управляющего воздействия.
PV	REAL		0.0	PROCESS VARIABLE / Значение переменной процесса На выходе "Значение переменной процесса" выводится эффективное значение переменной процесса.
ER	REAL		0.0	ERROR SIGNAL / Сигнал ошибки На выходе "Сигнал ошибки" выводится эффективное значение сигнала ошибки.

## 25.2 Пошаговое регулирование с помощью SFB42 / FB42 "CONT\_S"

### Введение

SFB42 / FB42 "CONT\_S" используется в программируемых логических контроллерах SIMATIC S7 для управления техническими процессами с помощью дискретных выходных сигналов управляющего воздействия для интегрирующих исполнительных звеньев. При назначении параметров Вы можете активировать или деактивировать частные функции ступенчатого ПИ-регулятора, чтобы настроить его на процесс. Вы можете легко выполнить назначения, используя утилиту для назначения параметров (из меню: **Start** [Пуск] > **Simatic** > **Step7** > **Assign PID Control parameters** [Назначение параметров ПИД-регулирования]). Интерактивное электронное справочное руководство можно найти в меню: **Start** [Пуск] > **Simatic** > **Step7** > **Assign PID Control English** [Назначение параметров ПИД-регулирования]).

### Применение

Этот регулятор можно использовать как ПИ-регулятор с постоянными уставками или во вторичных контурах регулирования в многоконтурных системах регулирования в качестве каскадного регулятора, регулятора состава смеси или регулятора соотношения, но не в качестве следящего регулятора. Принцип работы регулятора основан на ПИ-алгоритме управления дискретного регулятора, дополненном функциями для генерирования двоичного выходного сигнала из аналогового управляющего сигнала.

Следующее касается запуска с FB V1.5 или V1.1.0 для CPU 314 IFM:

При  $T_I = T\#0ms$  интегральный компонент регулятора может быть выключен, что позволяет использовать блок в качестве пропорционального регулятора.

Так как регулятор работает безо всякого сигнала позиционной обратной связи, рассчитанная управляющая переменная не будет точно соответствовать реальной позиции элемента управления процессом. Настройка выполняется, если управляющая переменная ( $ER \cdot GAIN$ ) становится отрицательной. Регулятор устанавливает выходной сигнал QLMNDN (нижнее значение управляющей переменной) пока не будет достигнут LMNR\_LS (нижний предел сигнала позиционной обратной связи).

Такой регулятор может быть также использован как вторичный в каскадном регуляторе. Вход уставки SP\_INT используется для назначения позиции для управляющего элемента процесса. В этом случае значение переменной процесса на входе и параметр TI ("время интегрирования") должен быть установлен в состояние 0.

Пример применения: регулирование температуры посредством регулирования нагрева с широтно-импульсным управлением и регулирования охлаждения с помощью вентиля. Для данного случая для полного закрывания вентиля управляющая переменная ( $GAIN \cdot ER$ ) должна иметь отрицательное значение.

## Описание

Наряду с функциями в цепи переменной процесса, SFB реализует готовый ПИ–регулятор с двоичным выходом управляющего воздействия и возможностью влияния на управляющее воздействие вручную. Ступенчатый регулятор действует без сигнала позиционной обратной связи. Ниже Вы найдете описание частных функций:

### Цепь уставки

Значение уставки вводится на входе **SP\_INT** в формате с плавающей точкой.

### Цепь переменной процесса

Значение переменной процесса может вводиться в формате периферии или в формате с плавающей точкой. Функция **CRP\_IN** преобразует периферийное значение **PV\_PER** в формат с плавающей точкой в диапазоне от -100 до +100 % в соответствии со следующей формулой:

$$\text{Выход CRP\_IN} = \text{PV\_PER} * \frac{100}{27648}$$

Функция **PV\_NORM** нормирует выход **CRP\_IN** в соответствии со следующей формулой:

$$\text{Выход PV\_NORM} = (\text{выход CRP\_IN}) * \text{PV\_FAC} + \text{PV\_OFF}$$

**PV\_FAC** по умолчанию равно 1, а **PV\_OFF** по умолчанию равно 0.

### Сигнал ошибки

Разность между значением уставки и значением переменной процесса называется сигналом ошибки. Для подавления малых незатухающих колебаний из-за квантованности регулирующего воздействия (например, из-за ограниченной разрешающей способности управляющего воздействия регулирующего клапана) к сигнал ошибки обрабатывается амплитудным фильтром с зоной нечувствительности (**DEADBAND**). Если **DEADB\_W = 0**, то амплитудный фильтр выключен.

### **ПИ-алгоритм для пошагового управления**

SFB работает без сигнала позиционной обратной связи. И-воздействие ПИ-алгоритма и расчетный сигнал позиционной обратной связи определяются в **одном** интеграторе (INT) и сравниваются с остающимся П-воздействием в качестве значения обратной связи. Разностный сигнал поступает в каскад трехпозиционного регулирования (THREE\_ST) и генератор импульсов (PULSEOUT), который вырабатывает импульсы для исполнительного устройства. Частота переключения регулятора может быть уменьшена настройкой порога в каскаде трехпозиционного регулирования.

### **Возмущающее воздействие**

Возмущающее воздействие может быть подано на вход **DISV**.

### **Инициализация**

SFB42 / FB42 "CONT\_S" имеет подпрограмму инициализации, которая прогоняется, когда установлен входной параметр COM\_RST = TRUE.

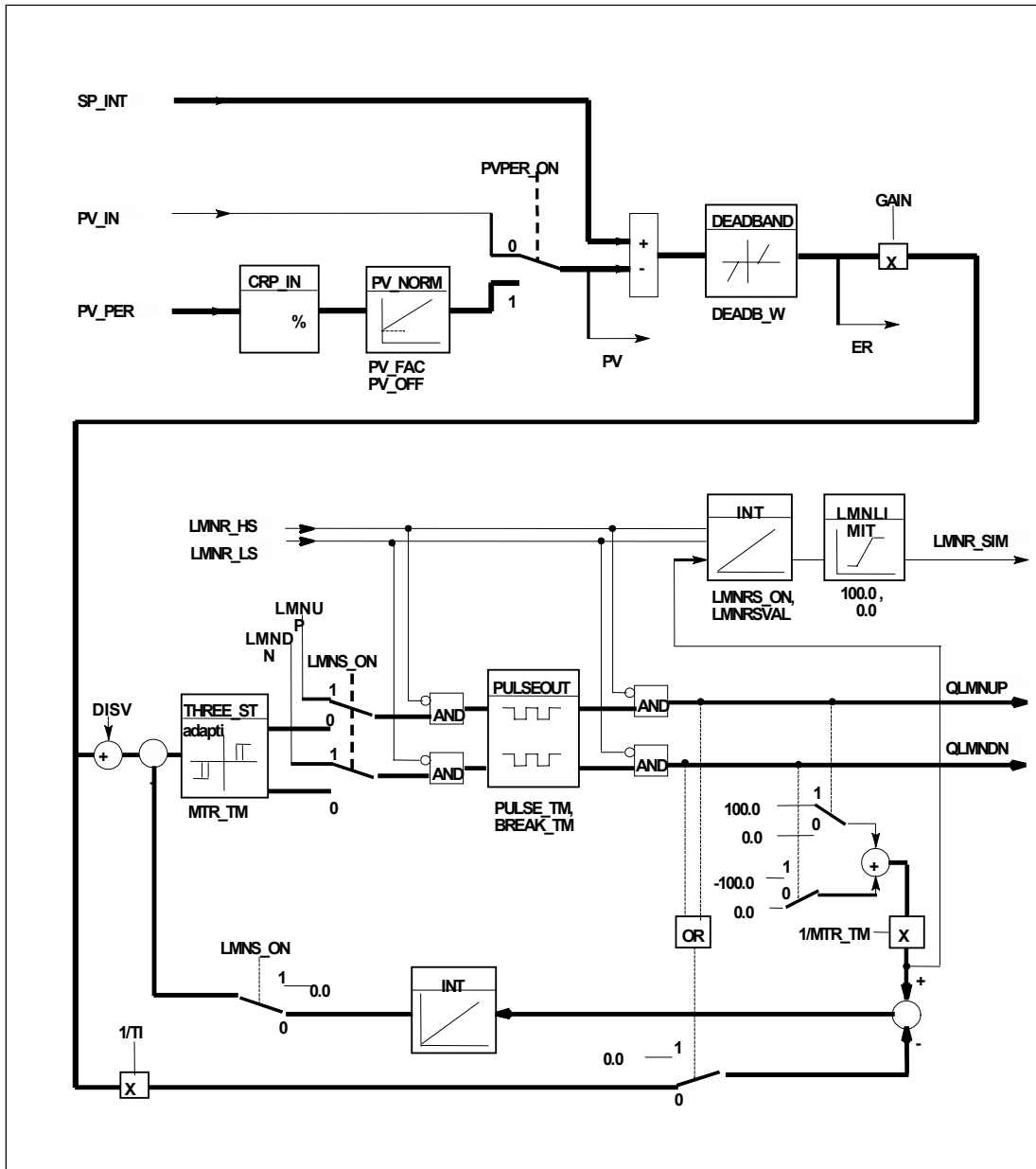
Все остальные выходы устанавливаются на их значения по умолчанию.

### **Информация об ошибках**

Параметр вывода ошибок RET\_VAL не используется.



Блок-схема регулятора пошагового управления



**Входные параметры**

Следующая таблица содержит описание входных параметров SFB42 "CONT\_S".

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
COM_RST	BOOL		FALSE	COMPLETE RESTART/ Полный рестарт Блок имеет подпрограмму инициализации, которая обрабатывается, когда установлен вход "COM_RST".
LMNR_HS	BOOL		FALSE	HIGH LIMIT OF POSITION FEEDBACK SIGNAL/ Верхняя граница сигнала позиционной обратной связи Сигнал "Исполнительное устройство у упора верхней границы" подключен к входу "Верхняя граница сигнала позиционной обратной связи". LMNR_HS=TRUE означает, что исполнительное устройство находится у упора верхней границы.
LMNR_LS	BOOL		FALSE	LOW LIMIT OF POSITION FEEDBACK SIGNAL/ Нижняя граница сигнала позиционной обратной связи Сигнал "Исполнительное устройство у упора нижней границы" подключен к входу "Нижняя граница сигнала позиционной обратной связи". LMNR_LS=TRUE означает, что исполнительное устройство находится у упора нижней границы.
LMNS_ON	BOOL		TRUE	MANUAL ACTUATING SIGNALS ON /Включение сигналов ручного воздействия Обработка управляющего сигнала включена на ручной режим на входе "Включение сигналов ручного воздействия".
LMNUP	BOOL		FALSE	ACTUATING SIGNALS UP / Увеличение управляющих сигналов В случае сигналов управляющего воздействия, вводимых вручную, выходной сигнал QLMNUP устанавливается на входе "Увеличение управляющих сигналов".
LMNDN	BOOL		FALSE	ACTUATING SIGNALS DOWN/ Уменьшение управляющих сигналов В случае сигналов управляющего воздействия, вводимых вручную, выходной сигнал QLMNDN устанавливается на входе "Уменьшение управляющих сигналов".
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERAL ON/ Включение переменной процесса периферии Если значение переменной процесса считывается из периферии, то вход PV_PER должен быть соединен с периферией, а вход "Включение переменной процесса периферии" должен быть установлен.
CYCLE	TIME	>= 1мс	T#1s	SAMPLING TIME/ Время опроса Время между вызовами блока должно быть постоянным. Вход "Время опроса" задает время между вызовами блока.

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
SP_INT	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>1)</sup>	0.0	INTERNAL SETPOINT/ Внутреннее значение уставки Вход "Внутреннее значение уставки" служит для установления заданного значения.
PV_IN	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>1)</sup>	0.0	PROCESS VARIABLE IN/ Ввод переменной процесса. На входе "Вход переменной процесса" может быть установлено значение инициализации или подключена внешняя переменная процесса в формате с плавающей точкой.
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERAL/ Переменная процесса от периферии Переменная процесса в формате периферии подключается к регулятору на входе "Переменная процесса от периферии".
GAIN	REAL		2.0	PROPORTIONAL GAIN/ Коэффициент пропорциональности Вход "Коэффициент пропорциональности" задает коэффициент усиления регулятора.
TI	TIME	>= CYCLE	T#20s	RESET TIME/ Время интегрирования Вход "Время интегрирования" определяет временную характеристику интегратора.
DEADB_W	REAL	0.0...100.0 (%) или физическая величина <sup>1)</sup>	1.0	DEAD BAND WIDTH/ Ширина зоны нечувствительности Зона нечувствительности соответствует ошибке. Вход "Ширина зоны нечувствительности" определяет размер зоны нечувствительности фильтра.
PV_FAC	REAL		1.0	PROCESS VARIABLE FACTOR / Коэффициент переменной процесса Вход "Коэффициент переменной процесса" умножается на значение переменной процесса. Вход служит для настройки диапазона переменной процесса.
PV_OFF	REAL		0.0	PROCESS VARIABLE OFFSET/ Смещение переменной процесса Вход "Смещение переменной процесса" складывается с фактическим значением. Вход служит для настройки диапазона переменной процесса.
PULSE_TM	TIME	>= CYCLE	T#3s	MINIMUM PULSE TIME /Минимальная длительность импульса Минимальная продолжительность импульса может быть установлена с помощью параметра "Минимальная длительность импульса".
BREAK_TM	TIME	>= CYCLE	T#3s	MINIMUM BREAK TIME / Минимальное время паузы Минимальная продолжительность паузы может быть установлена с помощью параметра "Минимальное время паузы".

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
MTR_TM	TIME	>= CYCLE	T#30s	MOTOR ACTUATING TIME / Время прогона привода Время, требуемое исполнительному устройству для перемещения от упора до упора, вводится в параметре "Время прогона привода".
DISV	REAL	От -100.0 до +100.0 (%) или физическая величина <sup>2)</sup>	0.0	DISTURBANCE VARIABLE/ Возмущающее воздействие Возмущающее воздействие подключается к входу "Возмущающее воздействие" для упреждающего регулирования.

1) Параметры в цепях уставки и переменной процесса с одинаковой единицей измерения

2) Параметры в цепи переменной управления с одинаковой единицей измерения

### Выходные параметры

Следующая таблица содержит описание выходных параметров SFB42 "CONT\_S".

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
QLMNUP	BOOL		FALSE	ACTUATING SIGNAL UP / Увеличение управляющего сигнала Если выход "Увеличение управляющего сигнала" установлен, то регулирующий клапан открыт.
QLMNDN	BOOL		FALSE	ACTUATING SIGNAL DOWN Уменьшение управляющего сигнала Если выход "Уменьшение управляющего сигнала" установлен, то регулирующий клапан закрыт.
PV	REAL		0.0	PROCESS VARIABLE / Переменная процесса На выходе "Переменная процесса" выводится эффективное значение переменной процесса.
ER	REAL		0.0	ERROR SIGNAL / Сигнал ошибки На выходе "Сигнал ошибки" выводится эффективное значение сигнала ошибки.

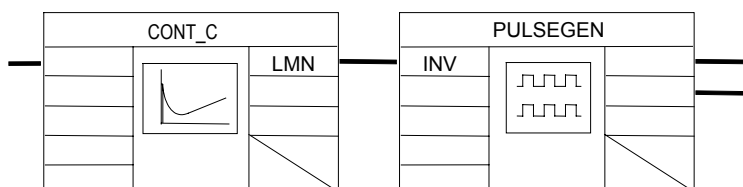
## 25.3 Формирование импульсов с помощью SFB43 / FB43 "PULSEGEN"

### Введение

SFB43 / FB43 "PULSEGEN" служит для построения ПИД-регулятора с импульсным выходом для пропорциональных исполнительных звеньев

### Применение

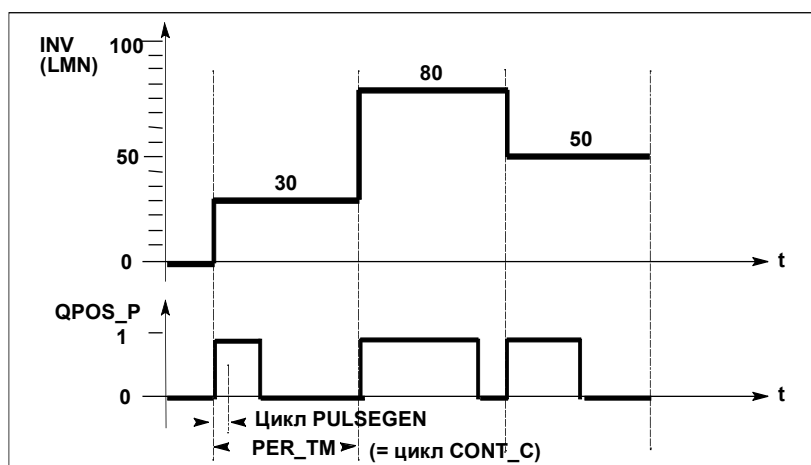
С помощью SFB "PULSEGEN" можно конфигурировать двух- или трехпозиционные ПИД-регуляторы с широтно-импульсной модуляцией. Эта функция обычно используется в соединении с непрерывным регулятором "CONT\_C".



### Описание

Функция PULSEGEN преобразует входную переменную INV (= управляющее воздействие ПИД-регулятора) путем модуляции длительности импульса в последовательность импульсов с постоянным периодом следования, который соответствует времени цикла обновления входной переменной и должен быть назначен в PER\_TM.

Длительность импульса на протяжении периода пропорциональна входной величине. Цикл, назначенный PER\_TM, не идентичен циклу обработки SFB "PULSEGEN." Цикл PER\_TM складывается из нескольких циклов обработки SFB "PULSEGEN," причем количество вызовов SFB "PULSEGEN" за цикл PER\_TM представляет собой меру точности широтно-импульсной модуляции.

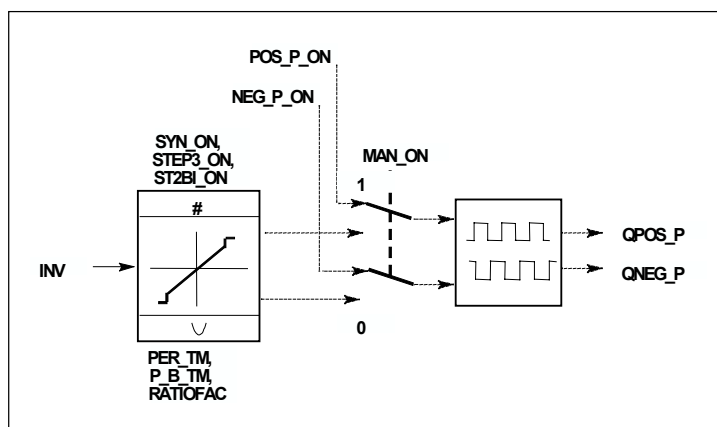


Широтно-импульсная модуляция

Входная величина 30% и 10 вызовов SFB / FB "PULSEGEN" на PER\_TM означает следующее:

- "единицу" на выходе QPOS для первых трех вызовов SFB / FB "PULSEGEN" (30% от 10 вызовов)
- "нуль" на выходе QPOS для семи последующих вызовов SFB / FB "PULSEGEN" (70% от 10 вызовов)

#### Блок-схема



#### Точность управляющего воздействия

В этом примере за счет "соотношения опроса" 1:10 (отношение количества вызовов CONT\_C к количеству вызовов PULSEGEN) точность управляющего воздействия ограничена 10%, то есть заданные входные значения INV могут быть имитированы длительностью импульсов на выходе QPOS только шагами в 10 %.

С увеличением количества вызовов SFB "PULSEGEN" на вызов CONT\_C точность повышается.

Например, если PULSEGEN вызывается в 100 раз чаще, чем CONT\_C, то достигается разрешение в 1% от диапазона управляющего воздействия.

#### Примечание

Частота вызовов должна программироваться пользователем.

### Автоматическая синхронизация

Существует возможность синхронизировать импульсный вывод с блоком, который обновляет входную переменную INV (например, CONT\_C). Это обеспечивает максимально быстрый вывод изменения входной переменной в виде импульса.

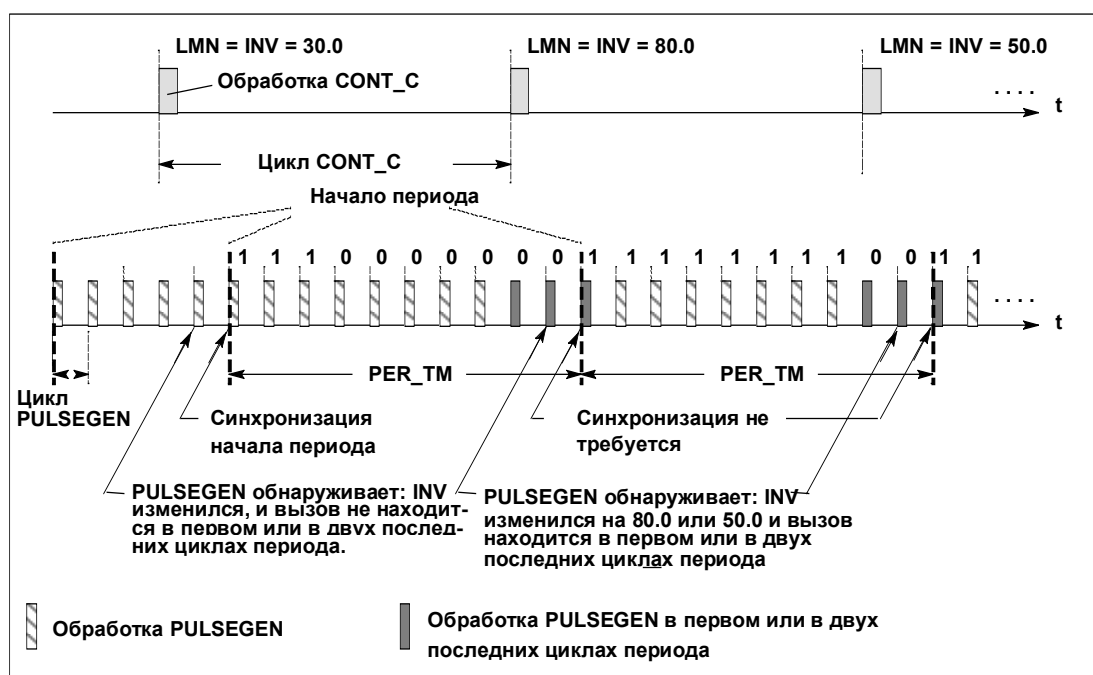
Формирователь импульсов оценивает входную величину на интервалах времени длиной в период PER\_TM и преобразует это значение в импульсный сигнал соответствующей длительности.

Однако, так как INV обычно рассчитывается в более медленном классе циклических прерываний, то формирователь импульсов должен начинать преобразование дискретного значения в импульсный сигнал как можно скорее после обновления INV.

Этот блок может синхронизировать начало периода с помощью следующей процедуры:

Если INV меняется и вызов блока не находится в первом или в двух последних циклах вызова периода, то проводится синхронизация.

Длительность импульса вычисляется вновь, и в следующем цикле вывод происходит с новым периодом.



Автоматическая синхронизация может быть заблокирована на входе "SYN\_ON" (= FALSE).

#### Примечание

В начале нового периода старое значение INV (то есть LMN) имитируется импульсным сигналом после синхронизации более или менее точно.

## Режимы работы

В зависимости от параметров, назначенных генератору импульсов, могут быть сконфигурированы ПИД-регуляторы с трехпозиционным выходом или с биполярным или униполярным двухпозиционным выходом. Следующая таблица иллюстрирует установку комбинаций выключателей для возможных режимов.

Режим	Ключ		
	MAN_ON	STEP3_ON	ST2BI_ON
Трехпозиционное регулирование	FALSE	TRUE	Любое
Двухпозиционное регулирование с биполярным диапазоном (от -100 % до +100 %)	FALSE	FALSE	TRUE
Двухпозиционное регулирование с униполярным диапазоном (0 % ...100 %)	FALSE	FALSE	FALSE
Ручной режим	TRUE	Любое	Любое

## Трехпозиционное регулирование

В режиме “Трехпозиционное регулирование” управляющий сигнал может принимать три состояния. Значения двоичных выходных сигналов QPOS\_P и QNEG\_P ставятся в соответствие состояниям исполнительного устройства. В таблице показан пример регулирования температуры:

Выходные сигналы	Исполнительное устройство		
	Нагрев	Выключено	Охлаждение
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

Исходя из входной величины, по характеристической кривой рассчитывается длительность импульса. Форма этой характеристической кривой определяется минимальной длительностью импульса или минимальной длительностью паузы и коэффициентом соотношения.

Нормальное значение коэффициента соотношения равно 1.

Точки излома на кривых определяются минимальной длительностью импульса или паузы.

### Минимальная длительность импульса или паузы

Правильно назначенная минимальная длительность импульса или паузы P\_V\_TM может предотвратить короткие времена включения или выключения, которые могут отрицательно влиять на срок службы переключательных звеньев и исполнительного оборудования.

### Примечание

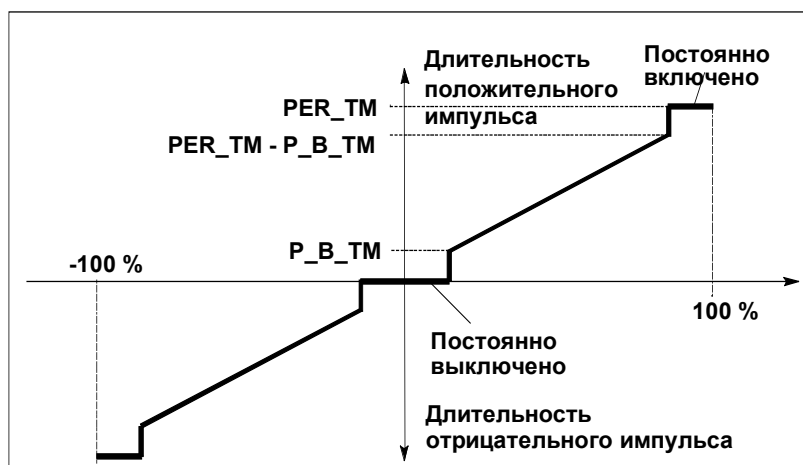
Малые абсолютные значения входной величины LMN, которые создавали бы длительность импульса меньше, чем P\_V\_TM, подавляются. Большие входные значения, которые создавали бы длительность импульса больше, чем (PER\_TM - P\_V\_TM), устанавливаются на 100 % или -100 %.



Длительность положительных или отрицательных импульсов рассчитывается из входной величины (в %), умноженной на длительность периода.

$$\text{Длительность импульса} = \frac{\text{INV}}{100} * \text{PER\_TM}$$

На следующем рисунке показана симметричная характеристика трехпозиционного регулятора (коэффициент соотношения = 1).



### Несимметричное трехпозиционное регулирование

С помощью коэффициента соотношения RATIOFAC можно изменять отношение длительности положительных импульсов к длительности отрицательных импульсов. Благодаря этому, например, в случае термического процесса можно учитывать разные постоянные времени объекта регулирования для нагревания и охлаждения.

Коэффициент соотношения влияет также на минимальную длительность импульса или паузы. Коэффициент соотношения < 1 означает, что пороговое значение для отрицательных импульсов умножается на этот коэффициент.

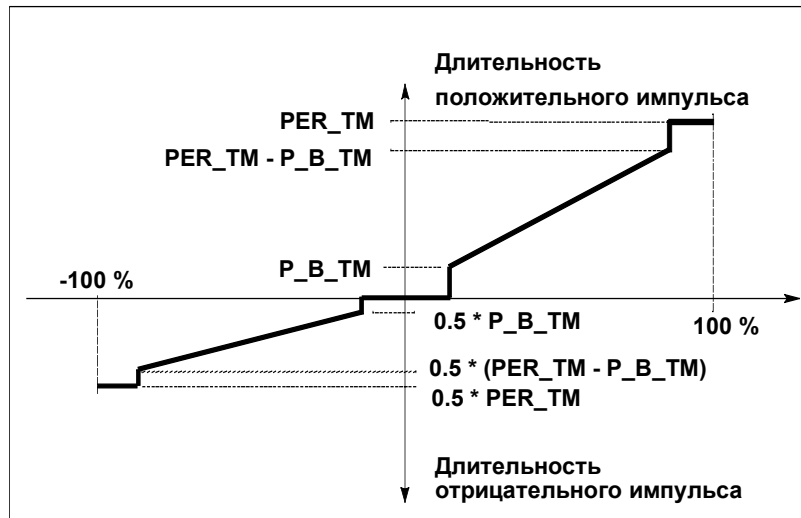
#### Коэффициент соотношения < 1

Длительность импульса на выходе отрицательных импульсов, рассчитанная из входной величины, умноженной на длительность периода, сокращается пропорционально коэффициенту соотношения (см. рисунок).

$$\text{Длительность положительного импульса} = \frac{\text{INV}}{100} * \text{PER\_TM}$$

$$\text{Длительность отрицательного импульса} = \frac{\text{INV}}{100} * \text{PER\_TM} * \text{RATIOFAC}$$

На следующем рисунке показана асимметричная характеристика трехступенчатого регулятора (коэффициент соотношения = 0.5):



#### Коэффициент соотношения > 1

Длительность импульса на выходе положительных импульсов, рассчитанная из входной величины, умноженной на длительность периода, изменяется обратно пропорционально коэффициенту соотношения.

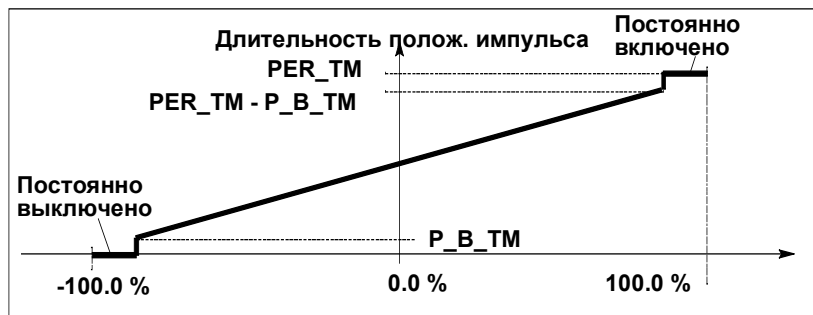
$$\text{Длительность отрицательного импульса} = \frac{\text{INV}}{100} * \text{PER\_TM}$$

$$\text{Длительность положительного импульса} = \frac{\text{INV}}{100} * \frac{\text{PER\_TM}}{\text{RATIOFAC}}$$

#### Двухпозиционное регулирование

При двухпозиционном регулировании к исполнительному устройству типа "включено-выключено" подключается только выход положительных импульсов QPOS\_P блока PULSEGEN. При двухпозиционном регулировании с соответствующим исполнительным звеном типа "включено-выключено" связывается только положительный импульсный выход QPOS\_P блока PULSEGEN. В зависимости от используемого диапазона управляющего воздействия двухпозиционный регулятор имеет биполярный или униполярный диапазон управляющего воздействия.

**Двухпозиционное регулирование с биполярным диапазоном управляющего воздействия (от -100% до 100%)**



**Двухпозиционное регулирование с униполярным диапазоном управляющего воздействия (от 0% до 100%)**



На QNEG\_P имеется в распоряжении инвертированный выходной сигнал на тот случай, когда включение двухпозиционного регулятора в контуре регулирования требует логически инвертированного двоичного сигнала для управляющих импульсов.

Импульс	Исполнительное устройство	
	Включено	Выключено
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

**Ручной режим при двух- или трехпозиционном регулировании**

В ручном режиме (MAN\_ON = TRUE) двоичные выходы трех- или двухпозиционного регулятора могут устанавливаться сигналами POS\_P\_ON и NEG\_P\_ON независимо от INV.

	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
Трехпозиционное регулирование	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
Двухпозиционное регулирование	FALSE	Any	FALSE	TRUE
	TRUE	Any	TRUE	FALSE

**Инициализация**

SFB42 / FB42 "PULSGEN" имеет подпрограмму инициализации, которая прогоняется, когда установлен входной параметр COM\_RST = TRUE.

Все выходы сигналов устанавливаются в 0.

**Информация об ошибках**

Параметр вывода ошибок RET\_VAL не используется.

**Входные параметры**

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
INV	REAL	-100.0...100.0 (%)	0.0	INPUT VARIABLE/ Входная переменная К входному параметру "Входная переменная" подключается аналоговое управляющее воздействие
PER_TM	TIME	>=20*CYCLE	T#1s	PERIOD TIME / Длительность периода С помощью входного параметра "Длительность периода" вводится постоянный период широтно-импульсной модуляции. Это соответствует времени опроса регулятора. Соотношение между временем опроса генератора импульсов и временем опроса регулятора определяет точность широтно-импульсной модуляции.
P_B_TM	TIME	>= CYCLE	T#0ms	MINIMUM PULSE/BREAK TIME / Минимальная длительность импульса/паузы Минимальная длительность импульса или паузы, которая может быть назначена во входных параметрах "Минимальная длительность импульса или паузы"
RATIOFAC	REAL	0.1 ...10.0	1.0	RATIO FACTOR / Коэффициент соотношения Входной параметр "Коэффициент соотношения" может быть использован для изменения отношения длительности отрицательного импульса к положительному. В термическом процессе это позволило бы, например, компенсировать разность постоянных времени нагрева и охлаждения (например, в процессе с электрическим нагревом и водяным охлаждением).

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
STEP3_ON	BOOL		TRUE	THREE STEP CONTROL ON / Включение трехпозиционного регулирования Входной параметр "Включение трехпозиционного регулирования" активирует этот режим. При трехпозиционном регулировании активны оба выходных сигнала.
ST2BI_ON	BOOL		FALSE	TWO STEP CONTROL FOR BIPOLAR MANIPULATED VALUE RANGE ON / Включение двухпозиционного регулирования для биполярного управляющего воздействия С помощью входного параметра "Включение двухпозиционного регулирования для биполярного управляющего воздействия" Вы можете выбирать между режимами "Двухпозиционное регулирование для биполярного управляющего воздействия" и "Двухпозиционное регулирование для униполярного управляющего воздействия". Должен быть установлен параметр STEP3_ON = FALSE.
MAN_ON	BOOL		FALSE	MANUAL MODE ON / Включение ручного режима При установке входного параметра "Включение ручного режима" выходные сигналы могут устанавливаться вручную.
POS_P_ON	BOOL		FALSE	POSITIVE PULSE ON / Включение положительного импульса В ручном режиме при трехпозиционном регулировании выходной сигнал QPOS_P может быть установлен через входной параметр "Включение положительного импульса". В ручном режиме при двухпозиционном регулировании, QNEG_P всегда устанавливается инверсно по отношению к QPOS_P.

Параметр	Тип данных	Диапазон значений	Значение по умолчанию	Описание
NEG_P_ON	BOOL		FALSE	NEGATIVE PULSE ON / Включение отрицательного импульса В ручном режиме при трехпозиционном регулировании выходной сигнал QNEG_P может быть установлен через входной параметр "Включение отрицательного импульса". В ручном режиме при двухпозиционном регулировании, QNEG_P всегда устанавливается инверсно по отношению к QPOS_P.
SYN_ON	BOOL		TRUE	SYNCHRONIZATION ON / Включение синхронизации При установке входного параметра "Включение синхронизации" имеется возможность автоматической синхронизации с блоком, обновляющим входную переменную INV. Это гарантирует максимально быстрый вывод изменения входной переменной в виде импульса.
COM_RST	BOOL		FALSE	COMPLETE RESTART / Полный рестарт Блок имеет подпрограмму инициализации, которая обрабатывается, когда установлен вход COM_RST.
CYCLE	TIME	>= 1мс	T#10ms	SAMPLING TIME / Длительность цикла опроса Время между вызовами блока должно быть постоянным. Вход "Длительность цикла опроса" определяет интервал между вызовами блока.

**Примечание**

Значения входных параметров в блоке не ограничиваются; проверка параметров не происходит.

**Выходные параметры**

Параметр	Тип данных	Значения	Значения по умолчанию	Описание
QPOS_P	BOOL		FALSE	OUTPUT POSITIVE PULSE / Выходной сигнал - положительный импульс Выходной параметр "Выходной сигнал - положительный импульс" устанавливается, когда должен выводиться импульс. При трехпозиционном регулировании это всегда положительный импульс. При двухпозиционном регулировании QNEG_P всегда устанавливается инверсно по отношению к QPOS_P.
QNEG_P	BOOL		FALSE	OUTPUT NEGATIVE PULSE / Выходной сигнал - отрицательный импульс Выходной параметр "Выходной сигнал - отрицательный импульс" устанавливается, когда должен выводиться импульс. При трехпозиционном регулировании это всегда отрицательный импульс. При двухпозиционном регулировании QNEG_P всегда устанавливается инверсно по отношению к QPOS_P.

## 25.4 Пример блока PULSEGEN

### Контур регулирования

С помощью регулятора непрерывного действия CONT\_C и импульсного генератора PULSEGEN Вы можете реализовать регулятор с фиксированной уставкой и переключающим выходом для пропорциональных исполнительных устройств. Следующий рисунок показывает поток сигналов в контуре регулирования.



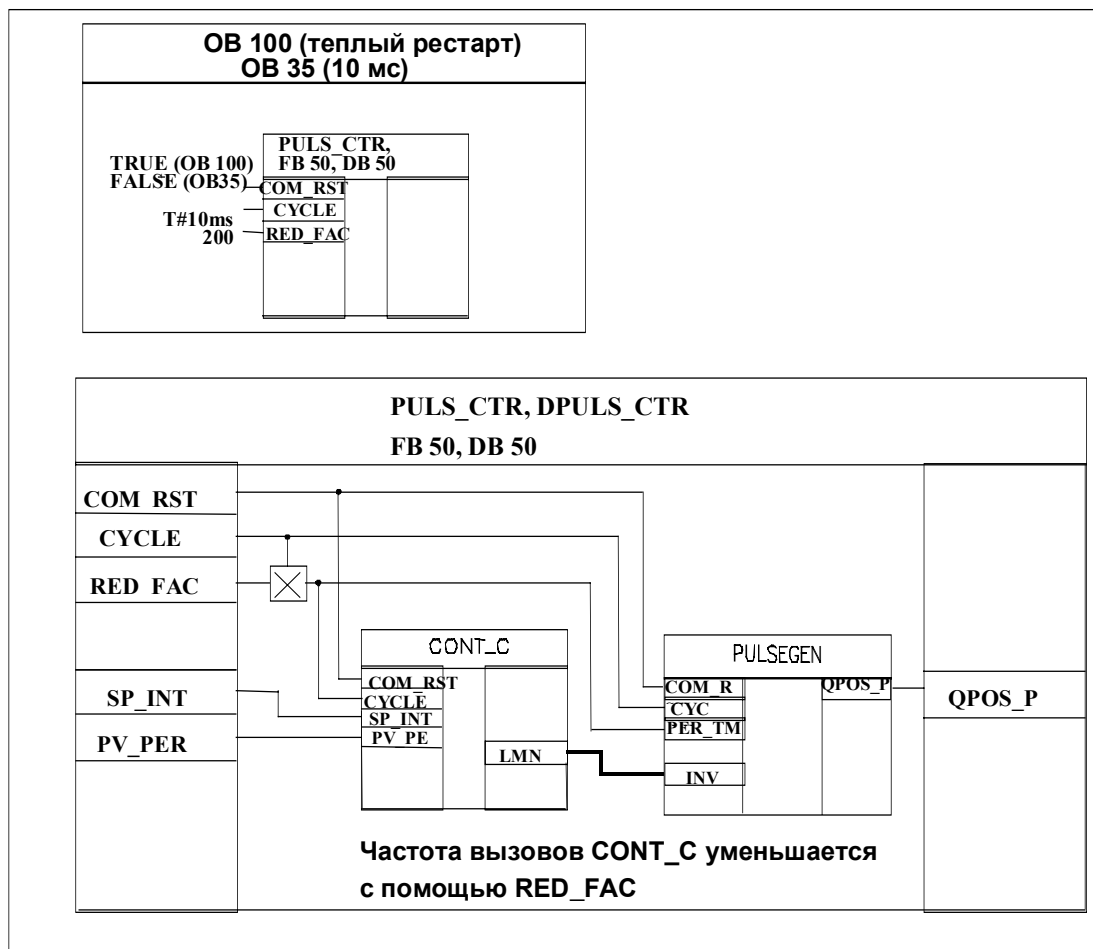
Регулятор непрерывного действия CONT\_C формирует управляющее воздействие LMN, которое преобразуется импульсным генератором PULSEGEN в импульсно-прерывистый сигнал QPOS\_P или QNEG\_P.

### Вызов блока и его подключение

Регулятор с фиксированной уставкой и переключающим выходом для пропорциональных исполнительных устройств PULS\_CTR состоит из блоков CONT\_C и PULSEGEN. Вызов блока осуществляется так, чтобы CONT\_C вызывался каждые 2 секунды ( $=\text{CYCLE} \cdot \text{RED\_FAC}$ ), а PULSEGEN каждые 10 мс ( $=\text{CYCLE}$ ). Время цикла OB35 устанавливается равным 10 мс. Взаимосвязь можно видеть на следующем рисунке.

Во время теплового рестарта в OB100 вызывается блок PULS\_CTR, и вход COM\_RST устанавливается в TRUE.





### Программа на STL для FB PULS\_CTR

Адрес	Описание	Имя	Тип	Комментарий
0.0	in	SP_INT	REAL	Уставка
4.0	in	PV_PER	WORD	Периферийное значение переменной процесса
6.0	in	RED_FAC	INT	Коэффициент уменьшения частоты вызовов
8.0	in	COM_RST	BOOL	Полный рестарт
10.0	in	CYCLE	TIME	Время опроса
14.0	out	QPOS_P	BOOL	Управляющий сигнал
16.0	stat	DI_CONT_C	FB-CONT_C	Счетчик
142.0	stat	DI_PULSEGEN	FB-PULSEGEN	Счетчик
176.0	stat	sCount	INT	Счетчик
0.0	temp	tCycCtr	TIME	Время опроса регулятора

STL	Описание
A #COM_RST	// Подпрограмма инициализации
JCN M001	
L 0	
T #sCount	
M001: L #CYCLE	//Вычисление цикла опроса регулятора
L #RED_FAC	
*D	
T #tCycCtr	
L #sCount	// Уменьшение значения счетчика и сравнение с нулем
L 1	
-I	
T #sCount	
L 0	
<=I	
JCN M002	//Условный вызов блока и установка счетчика
CALL #DI_CONT_C	
COM_RST :=#COM_RST	
CYCLE :=#tCycCtr	
SP_INT :=#SP_INT	
PV_PER :=#PV_PER	
L #RED_FAC	
T #sCount	
M002: L #DI_CONT_C.LMN	
T #DI_PULSEGEN.INV	
CALL #DI_PULSEGEN	
PER_TM :=#tCycCtr	
COM_RST :=#COM_RST	
CYCLE :=#CYCLE	
QPOS_P :=#QPOS_P	
BE	

## **26 SFB для компактных CPU**

## 26.1 Позиционирование с аналоговым выходом с использованием SFB 44 "Analog"

### Описание

Чтобы управлять функциями позиционирования из программы пользователя, используйте SFB "Analog" (SFB 44).

Функция обеспечивает стабилизированное заданное аналоговое значение напряжения от управляемого блока питания (**сигнал напряжения**)  $\pm 10$  V или тока (**сигнал тока**)  $\pm 20$  mA.

- После фазы ускорения (RAM\_UP) привод приближается к заданному положению с определенной скоростью ( $V_{\text{setpoint}}$ ).
- В "точке торможения", которая вычисляется CPU, включается заранее определенное замедление (RAMP\_DN) до "точки переключения".
- Когда "точка переключения" достигается, движение привода продолжается с малой скоростью ( $V_{\text{creep}}$ ).
- Привод выключается в "точке выключения".
- "Точка переключения" и "точка выключения" должны быть определены Вами для каждого "шага приближения" ("Step Approach") в значениях параметров **change-over difference** (приращение до переключения) и **cut-off difference** (приращение до выключения).  
Эти параметры могут быть определены отдельно для прямого движения (в положительном направлении) и для движения в обратную сторону (в отрицательном направлении).
- Движение завершается (**WORKING = FALSE** (ЛОЖЬ)) когда достигается "точка выключения".  
После этого может начаться выполнение нового задания по позиционированию привода.
- Заданная цель достигнута (**POS\_RCD = TRUE** (ИСТИНА)), когда значение фактического положения достигло **target range** ["целевого диапазона"]. Если фактическое положение привода медленно изменяется, без собственно выполнения задания по позиционированию привода, то сигнал "Position reached" ["Позиция достигнута"] не сбрасывается.

Если "приращение до переключения" меньше, чем "приращение до выключения", привод замедляет скорость от "точки торможения" до установленной скорости.

**Основные параметры:**

Здесь мы описываем параметры SFB, относящиеся ко всем рабочим режимам. Параметры, связанные с отдельными рабочими режимами описаны вместе с этими рабочими режимами.

**Параметры:**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>LADDR</b>	INPUT	WORD	0	Определяется типом CPU	W#16#0310	I/O адрес Вашего субмодуля, определенный в "HW Config". Если I и O адреса не равны, то младший из них должен быть определен.
<b>CHANNEL</b>	INPUT	INT	2	0	0	Номер канала
<b>STOP</b>	INPUT	BOOL	4.4	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Остановить выполнение. При STOP=TRUE (ИСТИНА) Вы можете остановить/прервать выполнение задания.
<b>ERR_A</b>	INPUT	BOOL	4.5	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Собирает подтверждения для внешних ошибок. Внешние ошибки квитируются с ERR_A = TRUE (ИСТИНА)
<b>SPEED</b>	INPUT	BOOL	12	Скорость "Стеер" до 1000000 имп/с но не выше максимальной скорости, объявленной в параметре	1000	Ось скорости до значения $V_{setpoint}$ . Невозможно изменить скорость во время выполнения задания.
<b>WORKING</b>	OUTPUT	BOOL	16.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Выполнение активировано
<b>ACT_POS</b>	OUTPUT	DINT	18	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Значение для фактической позиции
<b>MODE_OUT</b>	OUTPUT	INT	22	0, 1, 3, 4, 5	0	Активный/ сконфигурированный рабочий режим

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>ERR</b>	OUTPUT	WORD	24	"0" / "1"	0	Внешняя ошибка: Бит 2: мониторинг нулевой точки Бит 11: мониторинг прохождения диапазона (всегда 1) Бит 12: мониторинг рабочего режима. Бит 13: мониторинг фактического значения. Бит 14: мониторинг достижения точки назначения. Бит 15: мониторинг достижения зоны точки назначения. Остальные биты не используются
<b>ST_ENBLD</b>	OUTPUT	BOOL	26.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Параметр разрешения запуска CPU: старт разрешен, если все следующие условия выполнены: <ul style="list-style-type: none"> <li>• не ожидается переход к STOP (STOP = FALSE (ЛОЖЬ))</li> <li>• не ожидается внешняя ошибка (ERR = 0)</li> <li>• привод разблокирован (DRV_EN = TRUE (ИСТИНА))</li> <li>• задание на позиционирование привода не активировано (WORKING = FALSE (ЛОЖЬ))</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	26.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Ошибка при старте или при продолжении выполнения задания.
<b>STATUS</b>	OUTPUT	WORD	28.0	W#16#0000 ... W#16#FFFF	W#16#0000	Номер ошибки

**Параметры, неназначаемые в блоке (статические локальные данные)**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>ACCEL</b>	STATIC	DINT	30	1 ... 100 000 импульсов/с <sup>2</sup>	100	Ускорение. Изменение во время выполнения задания невозможно.
<b>DECEL</b>	STATIC	DINT	34	1 ... 100 000 импульсов/с <sup>2</sup>	100	Замедление. Изменение во время выполнения задания невозможно.
<b>CHGDIFF_P</b>	STATIC	DINT	38	0 ... 10 <sup>8</sup> импульсов	1000	Переключение "положительное приращение (плюс)": Определяет "точку переключения", после достижения которой движение привода вперед продолжается с малой скоростью ( $V_{creep}$ ).
<b>CUTOFF-DIFF_P</b>	STATIC	DINT	42	0 ... 10 <sup>8</sup> импульсов	100	Выключение "положительное приращение (плюс)": Определяет "точку выключения", после достижения которой движение привода вперед с малой скоростью ( $V_{creep}$ ) прекращается.
<b>CHGDIFF_M</b>	STATIC	DINT	46	0 ... 10 <sup>8</sup> импульсов	1000	Переключение "отрицательное приращение (минус)": Определяет "точку переключения", после достижения которой движение привода назад продолжается с малой скоростью ( $V_{creep}$ ).
<b>CUTOFF-DIFF_M</b>	STATIC	DINT	50	0 ... 10 <sup>8</sup> импульсов	100	Выключение "отрицательное приращение (минус)": Определяет "точку выключения", после достижения которой движение привода назад с малой скоростью ( $V_{creep}$ ) прекращается.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>PARA</b>	STATIC	BOOL	54.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметры назначены для направления (оси)
<b>DIR</b>	STATIC	BOOL	54.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Фактические/ последние данные датчика о направлении движения: FALSE (ЛОЖЬ) = вперед (положительное направление (плюс)); TRUE (ИСТИНА) = назад (отрицательное направление (минус)).
<b>CUTOFF</b>	STATIC	BOOL	54.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод в районе "точки выключения" (как в переходном состоянии от момента выключения к началу выполнения следующего перемещения).
<b>CHGOVER</b>	STATIC	BOOL	54.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод в районе "точки переключения" (как в переходном состоянии от момента достижения малой скорости ( $V_{creep}$ ) до начала выполнения следующего перемещения).
<b>RAMP_DN</b>	STATIC	BOOL	54.4	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод замедляет движение от "точки торможения" до "точки переключения".
<b>RAMP_UP</b>	STATIC	BOOL	54.5	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод ускоряет движение от "точки старта" до достижения скорости SPEED ( $V_{setpoint}$ ).
<b>DIST_TO_GO</b>	STATIC	DINT	56	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическая величина предстоящего перемещения привода.



Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
LAST_TRG	STATIC	DINT	60	$-5 \times 10^8 \dots$ $+5 \times 10^8$ импульсов	0	Последний/ текущий пункт назначения (цель) <ul style="list-style-type: none"> <li>• Absolute Step Approach (абсолютное пошаговое приближение): при запуске LAST_TRG = текущему абсолютному значению координаты пункта (позиции) назначения (TARGET).</li> <li>• Relative Step Approach (относительное пошаговое приближение): при запуске LAST_TRG = LAST_TRG (определяет +/- дистанцию с учетом предыдущей точки назначения (TARGET)).</li> </ul>

**Параметры для пошагового режима ("Jog") работы привода:**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Пошаговое движение привода в положительном направлении. (По переднему фронту сигнала).

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DIR_M</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Пошаговое движение привода в отрицательном направлении. (По переднему фронту сигнала).
<b>MODE_IN</b>	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 1 = пошаговое перемещение привода
<b>WORKING</b>	OUTPUT	BOOL	16.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.
<b>ACT_POS</b>	OUTPUT	DINT	18	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение положения привода
<b>MODE_OUT</b>	OUTPUT	INT	22	0, 1, 3, 4, 5	0	Активный/ сконфигурированный режим работы

**Параметры для режима относительного перемещения привода ("Reference run"):**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DRV_EN</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован
<b>DIR_P</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Относительное перемещение привода в положительном направлении. (По переднему фронту сигнала).
<b>DIR_M</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Относительное перемещение привода в отрицательном направлении. (По переднему фронту сигнала).
<b>MODE_IN</b>	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 3 = относительное перемещение привода
<b>WORKING</b>	OUTPUT	BOOL	16.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>SYNC</b>	OUTPUT	BOOL	16.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	SYNC = TRUE (ИСТИНА): направление (ось) синхронизировано.
<b>ACT_POS</b>	OUTPUT	DINT	18	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение положения привода
<b>MODE_OUT</b>	OUTPUT	INT	22	0, 1, 3, 4, 5	0	Активный/ сконфигурированный режим работы

**Параметры для режима "относительного пошагового приближения" ("Relative Step Approach"):**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DRV_EN</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован
<b>DIR_P</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в положительном направлении. (По переднему фронту сигнала).
<b>DIR_M</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в отрицательном направлении. (По переднему фронту сигнала).
<b>MODE_IN</b>	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 4 = режим "относительного пошагового приближения" ("Relative Step Approach") привода
<b>TARGET</b>	INPUT	DINT	8	$0 \dots 10^9$ импульсов	1000	Расстояние в импульсах (разрешены только положительные значения).
<b>WORKING</b>	OUTPUT	BOOL	16.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.
<b>POS_RCD</b>	OUTPUT	BOOL	16.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Позиция достигнута.

Параметр	Объявление	Тип данных	Адрес (экза-м-пляр DB)	Диапазон значений	Значение по умолчанию	Описание
ACT_POS	OUTPUT	DINT	18	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение положения привода
MODE_OUT	OUTPUT	INT	22	0, 1, 3, 4, 5	0	Фактический/ сконфигурированный режим работы

**Параметры для режима "абсолютного пошагового приближения" ("Absolute Step Approach"):**

Параметр	Объявление	Тип данных	Адрес (экза-м-пляр DB)	Диапазон значений	Значение по умолчанию	Описание
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован для доступа
START	INPUT	BOOL	4.1	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Включение перемещения привода. (По переднему фронту сигнала)
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в положительном направлении. (По переднему фронту сигнала)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в отрицательном направлении. (По переднему фронту сигнала)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 5 = режим "абсолютного пошагового приближения" ("Absolute Step Approach") привода
TARGET	INPUT	DINT	8	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов Ось вращения: 0 ... -1	1000	Расстояние до цели в импульсах.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>WORKING</b>	OUTPUT	BOOL	16.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.
<b>POS_RCD</b>	OUTPUT	BOOL	16.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Позиция достигнута.
<b>ACT_POS</b>	OUTPUT	DINT	18	$-5 \times 10^8$ ... $+5 \times 10^8$ импульсов	0	Фактическое значение положения привода
<b>MODE_OUT</b>	OUTPUT	INT	22	0, 1, 3, 4, 5	0	Фактический/ сконфигурированный режим работы

#### Параметры для задания "Установка опорной точки" ("Set Reference Point")

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>SYNC</b>	OUTPUT	BOOL	16.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	SYNC = TRUE (ИСТИНА): направление (ось) синхронизировано.

#### Параметры, неназначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>JOB_REQ</b>	STATIC	BOOL	76.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания. (По переднему фронту сигнала).
<b>JOB_DONE</b>	STATIC	BOOL	76.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Задание выполнено. Новое задание может стартовать.
<b>JOB_ERR</b>	STATIC	BOOL	76.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Отказ задания.
<b>JOB_ID</b>	STATIC	INT	78	1, 2	0	Задание: 1 = "Set Reference Point" (установка опорной точки)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
JOB_STAT	STATIC	WORD	80	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок при выполнении задания.
JOB_VAL	STATIC	DINT	82	$-5 \times 10^8$ ... $+5 \times 10^8$ импульсов	0	Параметр задания для координаты опорной точки.

### Параметры для задания "Отмена оставшегося перемещения" ("Clear Remaining Distance")

Параметры, незначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания. (По переднему фронту сигнала).
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	TRUE (ИСТИНА)	Задание выполнено. Новое задание может стартовать.
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Отказ задания.
JOB_ID	STATIC	INT	78	1, 2	<b>0</b>	Задание: 2 = "Clear Remaining Distance" (отмена оставшегося перемещения)
JOB_STAT	STATIC	WORD	80	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок при выполнении задания.
JOB_VAL	STATIC	DINT	82	-	0	Любое значение.

### Параметры для операции "Измерение длины" ("Length Measurement")

Эта операция начинается при приходе переднего фронта на дискретном входе. Она не имеет особых входных параметров.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
MSR_DONE	OUTPUT	BOOL	16.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Измерение длины закончено.

### Параметры, неназначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
BEG_VAL	STATIC	DINT	64	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение позиции в начале операции измерения.
END_VAL	STATIC	DINT	68	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение позиции по окончании операции измерения.
LEN_VAL	STATIC	DINT	72	$0 \dots 10^9$ импульсов	0	Измеренная длина.

### Информация об ошибках

#### Ошибки рабочего режима (ERROR = TRUE (ИСТИНА))

Если распознается ошибка, выходной параметр ERROR устанавливается в состояние TRUE (ИСТИНА). Параметр STATUS показывает причину ошибки.

Код ошибки	Объяснение
W#16#2002	Неверный SFB, используйте SFB 44.
W#16#2004	Неверный номер канала (CHANNEL). Установите канал "0".
W#16#3001	Задание прекращено из-за ошибки в задании при вызове одноименного SFB. Исправьте соответствующие параметры задания.

Код ошибки	Объяснение
W#16#3002	Изменение параметра MODE_IN не разрешено, пока привод активизирован. Дождитесь окончания текущего позиционирования привода.
W#16#3003	Неизвестный рабочий режим (MODE_IN). Допускаются значения: 1, 3, 4 и 5.
W#16#3004	Одновременно допускается только один запрос на начало выполнения задания. Разрешенные запросы на старт: DIR_P, DIR_M и START.
W#16#3005	START допускается только в режиме работы "Absolute Step Approach" (абсолютный шаг приближения). Запустите выполнение с DIR_P или DIR_M.
W#16#3006	DIR_P или DIR_M не разрешены для линейных осей (направлений) и для рабочего режима "Absolute Step Approach" (абсолютный шаг приближения). Запускайте задание посредством START.
W#16#3007	Оси не синхронизированы. Режим "Absolute Step Approach" (абсолютное пошаговое приближение) возможен только при синхронизации осей.
W#16#3008	Сбросьте рабочий диапазон. Вернитесь к выполнению задания только в пошаговом ("Jog") режиме.
W#16#3101	Запуск заблокирован, так как ось не имеет параметров. Задайте параметры в submodule "Positioning" посредством утилиты HW Config.
W#16#3102	Запуск заблокирован, так как заблокирован привод. Установите в SFB: DRV_EN = TRUE (ИСТИНА).
W#16#3103	Запуск заблокирован из-за режима STOP. Отмените STOP в SFB: STOP = FALSE (ЛОЖЬ).
W#16#3104	Запуск заблокирован, так как привод выполняет перемещение (позиционирование) (WORKING = TRUE (ИСТИНА)). Дождитесь окончания текущего позиционирования привода.
W#16#3105	Запуск заблокирован, так как по крайней мере одна ошибка ожидания не сброшена. Сначала устраните и сбросьте все внешние ошибки, затем перезапустите задание.
W#16#3202	Некорректное задание скорости в параметре SPEED. Заданная скорость выходит за допустимые пределы значений (до 1000 000 импульсов/с), хотя не превышает максимальное значение, заданное параметром.
W#16#3203	Заданное в параметре ACCEL ускорение выходит за допустимые пределы значений (1 ... 100 000 импульсов/с <sup>2</sup> ).
W#16#3204	Заданное в параметре DECEL ускорение выходит за допустимые пределы значений (1 ... 100 000 импульсов/с <sup>2</sup> ).
W#16#3206	Заданное значение скорости в параметре SPEED должно быть выше, чем или равно значению, заданному параметром.
W#16#3301	"Приращение до переключения" / "Приращение до выключения" слишком велико. Установите максимальное значение этих параметров: 10 <sup>8</sup> .
W#16#3304	"Приращение до выключения" слишком мало. "Приращение до выключения" должно иметь значение, равное, по крайней мере, половине величины "целевого диапазона".
W#16#3305	"Приращение до переключения" слишком мало. "Приращение до переключения" должно иметь значение, равное, по крайней мере, половине величины "целевого диапазона".
W#16#3401	Заданное значение "целевой" позиции привода находится вне рабочего диапазона. Для линейной оси и режима пошагового приближения (Step Approach) заданная целевая позиция должна быть внутри диапазона программных граничных переключателей (включительно).



Код ошибки	Объяснение
W#16#3402	Заданное значение "целевой" позиции привода некорректно. Для оси вращения заданная целевая позиция должна быть больше, чем 0 и меньше, чем конечное значение на оси вращения.
W#16#3403	Неверное значение расстояния для прохода привода. Значение расстояния для прохода привода при режиме относительного пошагового приближения (Relative Step Approach) должно быть положительно.
W#16#3404	Неверное значение расстояния для привода. Значение абсолютной координаты для привода должно быть больше, чем величина $(-5 \times 10^8)$ .
W#16#3405	Неверное значение расстояния для привода. Значение абсолютной координаты для привода должно быть меньше, чем величина $(+5 \times 10^8)$ .
W#16#3406	Неверное значение расстояния для привода. Значение абсолютной координаты для привода должно лежать внутри рабочего диапазона (+/- половина "целевого диапазона").
W#16#3501	Расстояние для прогона привода слишком велико. Значение координаты цели для привода + фактически оставшаяся дистанция прогона привода должна быть больше, чем или должна быть равна $(-5 \times 10^8)$ .
W#16#3502	Расстояние для прогона привода слишком велико. Значение координаты цели для привода + фактически оставшаяся дистанция прогона привода должна быть меньше, чем или должна быть равна $(+5 \times 10^8)$ .
W#16#3503	Расстояние для прогона привода слишком мало. Значение расстояния прогона привода в положительном направлении должно быть больше, чем заданное значение "Приращение до выключения" в положительном направлении.
W#16#3504	Расстояние для прогона привода слишком мало. Значение расстояния прогона привода в отрицательном направлении должно быть больше, чем заданное значение "Приращение до выключения" в отрицательном направлении.
W#16#3505	Расстояние для прогона привода слишком мало или граничный переключатель для положительного направления уже активирован. Последняя достигнутая целевая координата в положительном направлении (в рабочем диапазоне или на границе величины прогона привода) слишком близка к фактическому положению привода.
W#16#3506	Расстояние для прогона привода слишком мало или граничный переключатель для отрицательного направления уже активирован. Последняя достигнутая целевая координата в отрицательном направлении (в рабочем диапазоне или на границе величины прогона привода) слишком близка к фактическому положению привода.

### Ошибки выполнения задания (JOB\_ERR = TRUE (ИСТИНА))

Если распознается ошибка в процессе выполнения задания, выходной параметр JOB\_ERR устанавливается в состояние TRUE (ИСТИНА)). Параметр JOB\_STAT показывает причину ошибки.

Код ошибки	Объяснение
W#16#4001	Ось не имеет параметров. Задайте параметры в submodule "Positioning" посредством утилиты HW Config.
W#16#4002	Задание не выполняется, так как привод выполняет перемещение (позиционирование) (WORKING = TRUE (ИСТИНА)). Дождитесь окончания текущего позиционирования привода и повторите запуск задания.
W#16#4004	Неизвестное задание. Проверьте ID задания, исправьте и повторите запуск задания.
W#16#4101	Для линейной оси координатная опорная точка не может быть за пределами рабочего диапазона.
W#16#4102	Для линейной оси величина суммы (координата опорной точки + фактически оставшаяся дистанция прохода привода) должна быть больше или равна ( $-5 \times 10^8$ ).
W#16#4103	Для линейной оси величина суммы (координата опорной точки + фактически оставшаяся дистанция прохода привода) должна быть меньше или равна ( $+5 \times 10^8$ ).
W#16#4104	Для линейной оси величина суммы (координата опорной точки + фактическая дистанция до точки запуска) должна быть больше или равна ( $-5 \times 10^8$ ).
W#16#4105	Для линейной оси величина суммы (координата опорной точки + фактическая дистанция до точки запуска) должна быть меньше или равна ( $+5 \times 10^8$ ).
W#16#4106	Для оси вращения величина координаты опорной точки должна быть меньше, чем 0 и большей или равной конечному значению оси.

### Внешние ошибки (ERR)

Специальная схема контролирует выполнение, расстояние перемещения и подключенные периферийные устройства. Необходимо только, чтобы Вы включили текущий мониторинг в параметрах "Drive", "Axis" и "Encoder" в специальных экранных окнах пользовательского интерфейса.

Внешняя ошибка вызовет сообщение, когда она будет распознана контролирующим модулем. Внешние ошибки могут происходить независимо от запускаемых заданий. Вы должны всегда сбрасывать внешние ошибки посредством установки ERR\_A = TRUE (ИСТИНА).

Бит установки в SFB в параметре ERR (WORD) для внешних ошибок.

Мониторинг	Код ошибки	Номер бита в ERR-WORD
Нулевой импульс (Zero mark)	W#16#0004	2
Величина перемещения (прохода) привода	W#16#0800	11
Рабочий диапазон	W#16#1000	12

Мониторинг	Код ошибки	Номер бита в ERR-WORD
Фактическое значение	W#16#2000	13
Целевая позиция	W#16#4000	14
Целевой диапазон	W#16#8000	15

### Системные ошибки (System Error)

Системные ошибки индицируются с помощью BIE = FALSE (ЛОЖЬ).  
Системные ошибки вызываются ошибками, возникающими при чтении/записи экземпляров DB или множественными вызовами SFB.

## 26.2 Позиционирование для дискретного выхода с использованием SFB 46 "DIGITAL"

### Описание

Чтобы управлять функциями позиционирования с дискретным выходом с помощью программы пользователя, используйте **SFB 46 "DIGITAL"**. Функция обеспечивает четыре 24-вольтовых дискретных выхода, предназначенных для управления приводом. Эти выходы должны управлять силовым каскадом. В зависимости от выбранной конфигурации управления и режимов работы дискретные выходы обеспечивают управление направлением движения и уровнем скорости (большая (rapid) или малая (creep) скорость).

Расстояние измеряется с помощью двухфазного асимметричного 24-вольтового измерительного преобразователя.

- Сначала к целевой позиции привод приближается со скоростью ( $V_{\text{rapid}}$ ).
- В "точке переключения" скорость переключается на значение малой скорости ( $V_{\text{creep}}$ ).
- В "точке выключения" скорость переключается на значение 0. Привод останавливается.
- "Точка переключения" и "точка выключения" должны быть определены Вами для каждого "шага приближения" ("Step Approach") в значениях параметров **change-over difference** (приращение до переключения) и **cut-off difference** (приращение до выключения).  
Эти параметры могут быть определены отдельно для прямого движения (в положительном направлении) и для движения в обратную сторону (в отрицательном направлении).
- Движение завершается (**WORKING = FALSE (ЛОЖЬ)**) когда достигается "точка выключения".  
После этого может начаться новое выполнение задания по позиционированию привода.
- Заданная цель достигнута (**POS\_RCD = TRUE (ИСТИНА)**), когда значение фактического положения привода достигло **home target** ["искомой (целевой) позиции"]. Если фактическое положение привода медленно изменяется, без нового выполнения задания по позиционированию привода, то сигнал "Position reached" ["позиция достигнута"] не сбрасывается.

**Основные параметры:**

Здесь мы описываем параметры SFB, относящиеся ко всем рабочим режимам. Параметры, связанные с отдельными рабочими режимами описаны с этими рабочими режимами.

**Параметры:**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>LADDR</b>	INPUT	WORD	0	Определяется типом CPU	W#16#0310	I/O адрес Вашего субмодуля, определенный в "HW Config". Если I и O адреса не равны, то младший из них должен быть определен.
<b>CHANNEL</b>	INPUT	INT	2	0	0	Номер канала
<b>STOP</b>	INPUT	BOOL	4.4	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Остановить выполнение. При STOP=TRUE (ИСТИНА) Вы можете остановить/прервать выполнение задания.
<b>ERR_A</b>	INPUT	BOOL	4.5	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Собирает подтверждения для внешних ошибок. Внешние ошибки квитируются с ERR_A = TRUE (ИСТИНА)
<b>SPEED</b>	INPUT	BOOL	12.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Для скорости имеются два значения: большая (Rapid) и малая (Creep): TRUE (ИСТИНА) = режим "Rapid"; FALSE (ЛОЖЬ) = режим "Creep".
<b>WORKING</b>	OUTPUT	BOOL	14.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Выполнение активировано
<b>ACT_POS</b>	OUTPUT	DINT	16	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Значение для фактической позиции
<b>MODE_OUT</b>	OUTPUT	INT	20	0, 1, 3, 4, 5	0	Активный/сконфигурированный рабочий режим
<b>ERR</b>	OUTPUT	WORD	22	Каждый из битов	0	Внешняя ошибка: Бит 2: мониторинг

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
				"0" или "1"		нулевой точки Бит 11: мониторинг прохождения диапазона (всегда 1) Бит 12: мониторинг рабочего режима. Бит 13: мониторинг фактического значения. Бит 14: мониторинг достижения точки назначения. Бит 15: мониторинг достижения зоны точки назначения. Остальные биты не используются
<b>ST_ENBLD</b>	OUTPUT	BOOL	24.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Параметр разрешения запуска CPU: старт разрешен, если все следующие условия выполнены: <ul style="list-style-type: none"> <li>• не ожидается переход к STOP (STOP = FALSE (ЛОЖЬ))</li> <li>• не ожидается внешняя ошибка (ERR = 0)</li> <li>• привод разблокирован (DRV_EN = TRUE (ИСТИНА))</li> <li>• задание на позиционирование привода не активировано (WORKING = FALSE (ЛОЖЬ))</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	24.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Ошибка при старте или при продолжении выполнения задания.
<b>STATUS</b>	OUTPUT	WORD	26.0	W#16#0000 ... W#16#FFFF	W#16#0000	Номер ошибки

**Параметры, неназначаемые в блоке (статические локальные данные)**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>CHGDIFF_P</b>	STATIC	DINT	28	0 ... 10 <sup>8</sup> импульсов	1000	Переключение "положительное приращение (плюс)": Определяет "точку переключения", после достижения которой движение привода вперед продолжается с малой скоростью ( $V_{creep}$ ).
<b>CUTOFF-DIFF_P</b>	STATIC	DINT	32	0 ... 10 <sup>8</sup> импульсов	100	Выключение "положительное приращение (плюс)": Определяет "точку выключения", после достижения которой движение привода вперед с малой скоростью ( $V_{creep}$ ) прекращается.
<b>CHGDIFF_M</b>	STATIC	DINT	36	0 ... 10 <sup>8</sup> импульсов	1000	Переключение "отрицательное приращение (минус)": Определяет "точку переключения", после достижения которой движение привода назад продолжается с малой скоростью ( $V_{creep}$ ).
<b>CUTOFF-DIFF_M</b>	STATIC	DINT	40	0 ... 10 <sup>8</sup> импульсов	100	Выключение "отрицательное приращение (минус)": Определяет "точку выключения", после достижения которой движение привода назад с малой скоростью ( $V_{creep}$ ) прекращается.
<b>PARA</b>	STATIC	BOOL	44.0	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметры назначены для направления (оси)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DIR</b>	STATIC	BOOL	44.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Фактические/ последние данные датчика о направлении движения: FALSE (ЛОЖЬ) = вперед (положительное направление (плюс)); TRUE (ИСТИНА) = назад (отрицательное направление (минус)).
<b>CUTOFF</b>	STATIC	BOOL	44.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод в районе "точки выключения" (как в переходном состоянии от момента выключения к началу выполнения следующего перемещения).
<b>CHGOVER</b>	STATIC	BOOL	44.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод в районе "точки переключения" (как в переходном состоянии от момента достижения малой скорости ( $V_{creep}$ ) до начала выполнения следующего перемещения).
<b>DIST_TO_GO</b>	STATIC	DINT	46	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическая величина оставшегося перемещения привода.



Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>LAST_TRG</b>	STATIC	DINT	50	$-5 \times 10^8 \dots$ $+5 \times 10^8$ импульсов	0	Последний/ текущий пункт назначения (цель) <ul style="list-style-type: none"> <li>• Absolute Step Approach (абсолютное пошаговое приближение): при запуске LAST_TRG = текущему абсолютному значению координаты пункта (позиции) назначения (TARGET).</li> <li>• Relative Step Approach (относительное пошаговое приближение): при запуске LAST_TRG = LAST_TRG (определяет +/- дистанцию с учетом предыдущей точки назначения (TARGET)).</li> </ul>

#### Параметры для пошагового режима ("Jog") работы привода:

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DRV_EN</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован
<b>DIR_P</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Пошаговое движение привода в положительном направлении. (По переднему фронту сигнала).

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DIR_M</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Пошаговое движение привода в отрицательном направлении. (По переднему фронту сигнала).
<b>MODE_IN</b>	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 1 = пошаговое перемещение привода
<b>WORKING</b>	OUTPUT	BOOL	14.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.
<b>ACT_POS</b>	OUTPUT	DINT	16	-5x10 <sup>8</sup> ... +5x10 <sup>8</sup> импульсов	0	Фактическое значение положения привода
<b>MODE_OUT</b>	OUTPUT	INT	20	0, 1, 3, 4, 5	0	Активный/ сконфигурированный режим работы

**Параметры для режима относительного перемещения привода ("Reference run"):**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DRV_EN</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован
<b>DIR_P</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Относительное перемещение привода в положительном направлении. (По переднему фронту сигнала).
<b>DIR_M</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Относительное перемещение привода в отрицательном направлении. (По переднему фронту сигнала).
<b>MODE_IN</b>	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 3 = относительное перемещение привода
<b>WORKING</b>	OUTPUT	BOOL	14.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>SYNC</b>	OUTPUT	BOOL	14.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	SYNC = TRUE (ИСТИНА): направление (ось) синхронизировано.
<b>ACT_POS</b>	OUTPUT	DINT	16	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение положения привода
<b>MODE_OUT</b>	OUTPUT	INT	20	0, 1, 3, 4, 5	0	Активный/ сконфигурированный режим работы

**Параметры для режима "относительного пошагового приближения" ("Relative Step Approach"):**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>DRV_EN</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован
<b>DIR_P</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в положительном направлении. (По переднему фронту сигнала).
<b>DIR_M</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в отрицательном направлении. (По переднему фронту сигнала).
<b>MODE_IN</b>	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 4 = режим "относительного пошагового приближения" ("Relative Step Approach") привода
<b>TARGET</b>	INPUT	DINT	8	$0 \dots 10^9$ импульсов	1000	Расстояние в импульсах (разрешены только положительные значения).
<b>WORKING</b>	OUTPUT	BOOL	14.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.
<b>POS_RCD</b>	OUTPUT	BOOL	14.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Позиция достигнута.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
ACT_POS	OUTPUT	DINT	16	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение положения привода
MODE_OUT	OUTPUT	INT	20	0, 1, 3, 4, 5	0	Активный/ сконфигурированный режим работы

**Параметры для режима "абсолютного пошагового приближения" ("Absolute Step Approach"):**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
DRV_EN	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Привод разблокирован для доступа
START	INPUT	BOOL	4.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Включение перемещения привода. (По переднему фронту сигнала)
DIR_P	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в положительном направлении. (По переднему фронту сигнала)
DIR_M	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Перемещение привода в отрицательном направлении. (По переднему фронту сигнала)
MODE_IN	INPUT	INT	6	0, 1, 3, 4, 5	1	Рабочий режим: 5 = режим "абсолютного пошагового приближения" ("Absolute Step Approach") привода
TARGET	INPUT	DINT	8	Линейная ось: $-5 \times 10^8 \dots +5 \times 10^8$ импульсов Ось вращения: 0 ... -1	1000	Расстояние до цели в импульсах.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>WORKING</b>	OUTPUT	BOOL	14.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние задания.
<b>POS_RCD</b>	OUTPUT	BOOL	14.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Позиция достигнута.
<b>ACT_POS</b>	OUTPUT	DINT	16	$-5 \times 10^8$ ... $+5 \times 10^8$ импульсов	0	Фактическое значение положения привода
<b>MODE_OUT</b>	OUTPUT	INT	20	0, 1, 3, 4, 5	0	Активный/ сконфигурированный режим работы

#### Параметры для задания "Установка опорной точки" ("Set Reference Point")

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>SYNC</b>	OUTPUT	BOOL	14.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	SYNC = TRUE (ИСТИНА): направление (ось) синхронизировано.

#### Параметры, неназначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>JOB_REQ</b>	STATIC	BOOL	66.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания. (По переднему фронту сигнала).
<b>JOB_DONE</b>	STATIC	BOOL	66.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Задание выполнено. Новое задание может стартовать.
<b>JOB_ERR</b>	STATIC	BOOL	66.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Отказ задания.
<b>JOB_ID</b>	STATIC	INT	68	1, 2	0	Задание: 1 = "Set Reference Point" (установка опорной точки)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
JOB_STAT	STATIC	WORD	70	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок при выполнении задания.
JOB_VAL	STATIC	DINT	72	$-5 \times 10^8$ ... $+5 \times 10^8$ импульсов	0	Параметр задания для координаты опорной точки.

### Параметры для задания "Отмена оставшегося перемещения" ("Clear Remaining Distance")

Параметры, незначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания. (По переднему фронту сигнала).
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	TRUE (ИСТИНА)	Задание выполнено. Новое задание может стартовать.
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Отказ задания.
JOB_ID	STATIC	INT	68	1, 2	<b>0</b>	Задание: 2 = "Clear Remaining Distance" (отмена оставшегося перемещения)
JOB_STAT	STATIC	WORD	70	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок при выполнении задания.
JOB_VAL	STATIC	DINT	72	-	0	Любое значение.

### Параметры для операции "Измерение длины" ("Length Measurement")

Эта операция начинается при приходе переднего фронта на дискретном входе. Она не имеет особых входных параметров.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
MSR_DONE	OUTPUT	BOOL	14.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Измерение длины закончено.

### Параметры, неназначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
BEG_VAL	STATIC	DINT	54	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение позиции в начале операции измерения.
END_VAL	STATIC	DINT	58	$-5 \times 10^8 \dots +5 \times 10^8$ импульсов	0	Фактическое значение позиции по окончании операции измерения.
LEN_VAL	STATIC	DINT	62	$0 \dots 10^9$ импульсов	0	Измеренная длина.

### Информация об ошибках

#### Ошибки рабочего режима (ERROR = TRUE (ИСТИНА))

Если распознается ошибка, выходной параметр ERROR устанавливается в состояние TRUE (ИСТИНА). Параметр STATUS показывает причину ошибки.

Код ошибки	Объяснение
W#16#2001	Неверный SFB, используйте SFB 46.
W#16#2004	Неверный номер канала (CHANNEL). Установите канал "0".

Код ошибки	Объяснение
W#16#3001	Задание прекращено из-за ошибки в задании при вызове одноименного SFB. Исправьте соответствующие параметры задания.
W#16#3002	Изменение параметра MODE_IN не разрешено, пока привод активизирован. Дождитесь окончания текущего позиционирования привода.
W#16#3003	Неизвестный рабочий режим (MODE_IN). Допускаются значения 1,3,4 и 5.
W#16#3004	Одновременно допускается только один запрос на начало выполнения задания. Разрешенные запросы на старт: DIR_P, DIR_M и START.
W#16#3005	START допускается только в режиме работы "Absolute Step Approach" (абсолютный шаг приближения). Запустите выполнение с DIR_P или DIR_M.
W#16#3006	DIR_P или DIR_M не разрешены для линейных осей (направлений) и для рабочего режима "Absolute Step Approach" (абсолютное пошаговое приближение). Запускайте задание посредством START.
W#16#3007	Оси не синхронизированы. Режим "Absolute Step Approach" (абсолютное пошаговое приближение) возможен только при синхронизации осей.
W#16#3008	Сбросьте рабочий диапазон. Вернитесь к выполнению задания только в пошаговом ("Jog") режиме.
W#16#3101	Запуск заблокирован, так как ось не имеет параметров. Задайте параметры в submodule "Positioning" посредством утилиты HW Config.
W#16#3102	Запуск заблокирован, так как заблокирован привод. Установите в SFB: DRV_EN = TRUE (ИСТИНА).
W#16#3103	Запуск заблокирован из-за режима STOP. Отмените STOP в SFB: STOP = FALSE (ЛОЖЬ).
W#16#3104	Запуск заблокирован, так как привод выполняет перемещение (позиционирование) (WORKING = TRUE (ИСТИНА)). Дождитесь окончания текущего позиционирования привода.
W#16#3105	Запуск заблокирован, так как по крайней мере одна ошибка ожидания не сброшена. Сначала устраните и сбросьте все внешние ошибки, затем перезапустите задание.
W#16#3201	Некорректное задание скорости в параметре SPEED. Для позиционирования с дискретными выходами управления скорость может иметь два допустимых значения: "Creep" (0) и "Rapid" (1).
W#16#3301	"Приращение до переключения" / "Приращение до выключения" слишком велико. Установите максимальное значение этих параметров: $10^3$ .
W#16#3303	"Приращение до переключения" слишком мало. "Приращение до переключения" должно иметь значение, по крайней мере не меньшее, чем величина "приращения до выключения".
W#16#3304	"Приращение до выключения" слишком мало. "Приращение до выключения" должно иметь значение, равное, по крайней мере, половине величины "целевого диапазона".
W#16#3401	Заданное значение "целевой" позиции привода находится вне рабочего диапазона. Для линейной оси и режима пошагового приближения (Step Approach) заданная целевая позиция должна быть внутри диапазона программных граничных переключателей (включительно).



Код ошибки	Объяснение
W#16#3402	Заданное значение "целевой" позиции привода некорректно. Для оси вращения заданная целевая позиция должна быть больше, чем 0 и меньше, чем конечное значение на оси вращения.
W#16#3403	Неверное значение расстояния для прохода привода. Значение расстояния для прохода привода при режиме относительного пошагового приближения (Relative Step Approach) должно быть положительно.
W#16#3404	Неверное значение расстояния для привода. Значение абсолютной координаты для привода должно быть больше, чем величина $(-5 \times 10^8)$ .
W#16#3405	Неверное значение расстояния для привода. Значение абсолютной координаты для привода должно быть меньше, чем величина $(+5 \times 10^8)$ .
W#16#3406	Неверное значение расстояния для привода. Значение абсолютной координаты для привода должно лежать внутри рабочего диапазона (+/- половина "целевого диапазона").
W#16#3501	Расстояние для прогона привода слишком велико. Значение координаты цели для привода + фактически оставшаяся дистанция прогона привода должна быть больше, чем или должна быть равна $(-5 \times 10^8)$ .
W#16#3502	Расстояние для прогона привода слишком велико. Значение координаты цели для привода + фактически оставшаяся дистанция прогона привода должна быть меньше, чем или должна быть равна $(+5 \times 10^8)$ .
W#16#3503	Расстояние для прогона привода слишком мало. Значение расстояния прогона привода в положительном направлении должно быть больше, чем определенное значение "Приращение до выключения" в положительном направлении.
W#16#3504	Расстояние для прогона привода слишком мало. Значение расстояния прогона привода в отрицательном направлении должно быть больше, чем определенное значение "Приращение до выключения" в отрицательном направлении.
W#16#3505	Расстояние для прогона привода слишком мало или граничный переключатель для положительного направления уже активирован. Последняя достигнутая целевая координата в положительном направлении (в рабочем диапазоне или на границе величины прогона привода) слишком близка к фактическому положению привода.
W#16#3506	Расстояние для прогона привода слишком мало или граничный переключатель для отрицательного направления уже активирован. Последняя достигнутая целевая координата в отрицательном направлении (в рабочем диапазоне или на границе величины прогона привода) слишком близка к фактическому положению привода.

### Ошибки выполнения задания (JOB\_ERR = TRUE (ИСТИНА))

Если распознается ошибка в процессе выполнения задания, выходной параметр JOB\_ERR устанавливается в состояние TRUE (ИСТИНА)). Параметр JOB\_STAT показывает причину ошибки.

Код ошибки	Объяснение
W#16#4001	Ось не имеет параметров. Задайте параметры в submodule "Positioning" посредством утилиты HW Config.
W#16#4002	Задание не выполняется, так как привод выполняет перемещение (позиционирование) (WORKING = TRUE (ИСТИНА)). Дождитесь окончания текущего позиционирования привода и повторите запуск задания.
W#16#4004	Неизвестное задание. Проверьте ID задания, исправьте и повторите запуск задания.
W#16#4101	Для линейной оси координатная опорная точка не может быть за пределами рабочего диапазона.
W#16#4102	Для линейной оси величина суммы (координата опорной точки + фактически оставшаяся дистанция прохода привода) должна быть больше или равна $(-5 \times 10^8)$ .
W#16#4103	Для линейной оси величина суммы (координата опорной точки + фактически оставшаяся дистанция прохода привода) должна быть меньше или равна $(+5 \times 10^8)$ .
W#16#4104	Для линейной оси величина суммы (координата опорной точки + фактическая дистанция до точки запуска) должна быть больше или равна $(-5 \times 10^8)$ .
W#16#4105	Для линейной оси величина суммы (координата опорной точки + фактическая дистанция до точки запуска) должна быть меньше или равна $(+5 \times 10^8)$ .
W#16#4106	Для оси вращения величина координаты опорной точки должна быть меньше, чем 0 и большей или равной конечному значению оси.

### Внешние ошибки (ERR)

Специальная схема контролирует выполнение, расстояние перемещения и подключенные периферийные устройства. Необходимо только, чтобы Вы включили текущий мониторинг в параметрах "Drive", "Axis" и "Encoder" в специальных экранных окнах пользовательского интерфейса.

Внешняя ошибка вызовет сообщение, когда она будет распознана контролирующим модулем. Внешние ошибки могут происходить независимо от запускаемых заданий. Вы должны всегда сбрасывать внешние ошибки посредством установки ERR\_A = TRUE (ИСТИНА).

Бит установки в SFB в параметре ERR (WORD) для внешних ошибок.

Мониторинг	Код ошибки	Номер бита в ERR-WORD
Нулевой импульс (Zero mark)	W#16#0004	2
Величина перемещения (прохода) привода	W#16#0800	11
Рабочий диапазон	W#16#1000	12
Фактическое значение	W#16#2000	13
Целевая позиция	W#16#4000	14

Мониторинг	Код ошибки	Номер бита в ERR-WORD
Целевой диапазон	W#16#8000	15

### Системные ошибки (System Error)

Системные ошибки индицируются с помощью BIE = FALSE (ЛОЖЬ).  
Системные ошибки вызываются ошибками, возникающими при чтении/записи экземпляров DB или множественными вызовами SFB.

## 26.3 Управление счетчиком с использованием SFB 47 "COUNT"

### Описание

Чтобы управлять счетчиком с помощью программы пользователя, используйте **SFB 47 "COUNT"**.

Функция обеспечивает следующие операции:

- Запуск / остановка счетчика через программный шлюз **SW\_GATE**.
- Включение / управление выходом DO.
- Восстановление состояния битов **STS\_CMP**, **STS\_OFLW**, **STS\_UFLW** и **STS\_ZP**.
- Восстановление фактического значения счетчика **COUNTVAL**.
- Задания чтения / записи во внутренних регистрах счетчика.

### Параметры:

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>LADDR</b>	INPUT	WORD	0	Определяется типом CPU	W#16#0300	I/O адрес Вашего submodule, определенный в "HW Config". Если I и O адреса не равны, то младший из них должен быть определен.
<b>CHANNEL</b>	INPUT	INT	2	CPU 312C: 0 ... 1 CPU 313C: 0 ... 2 CPU 314C: 0 ... 3	0	Номер канала
<b>SW_GATE</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Программный шлюз для запуска и остановки счетчика.
<b>CTRL_DO</b>	INPUT	BOOL	4.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Разрешение ручного управления выходом
<b>SET_DO</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управление выходом
<b>JOB_REQ</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания (по переднему фронту сигнала).

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>JOB_ID</b>	INPUT	WORD	6	W#16#0000 задание без функции W#16#0001 запись значения счетчика W#16#0002 запись загруженного значения W#16#0004 запись результата сравнения W#16#0008 запись значения гистерезиса W#16#0010 запись периода импульсов W#16#0082 чтение загруженного значения W#16#0084 чтение результата сравнения W#16#0088 чтение значения гистерезиса W#16#0090 чтение периода импульсов	W#16#0000	Номер задания
<b>JOB_VAL</b>	INPUT	DINT	8	$-2^{31}$ ... $(+2^{31} - 1)$	0	Значение для задания записи
<b>STS_GATE</b>	OUTPUT	BOOL	12.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние внутреннего шлюза
<b>STS_STRT</b>	OUTPUT	BOOL	12.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние программного шлюза (вход запуска)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
STS_LTCH	OUTPUT	BOOL	12.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние входа защелки
STS_DO	OUTPUT	BOOL	12.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние выхода
STS_C_DN	OUTPUT	BOOL	12.4	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние обратного счета счетчика. Всегда отображает направление последнего счета. Значение STS_C_DN равно FALSE (ЛОЖЬ) после первого вызова SFB.
STS_C_UP	OUTPUT	BOOL	12.5	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние прямого счета счетчика. Всегда отображает направление последнего счета. Значение STS_C_UP равно TRUE (ИСТИНА) после первого вызова SFB.
COUNTVAL	OUTPUT	DINT	14	$-2^{31}$ ... $(+2^{31} - 1)$	0	Фактическое значение счетчика
LATCHVAL	OUTPUT	DINT	18	$-2^{31}$ ... $(+2^{31} - 1)$	0	Фактическое значение защелки
JOB_DONE	OUTPUT	BOOL	22.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Новое задание может стартовать
JOB_ERR	OUTPUT	BOOL	22.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Сбой задания
JOB_STAT	OUTPUT	WORD	24	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок задания

#### Примечание

Если Вы установили параметр "Reaction of the output" (реакция выходного значения) в состояние "No comparison" (не сравнивать), то справедливо следующее:

- Выход будет включен как обычный выход.
- Входные параметры CTRL\_DO и SET\_DO блока SFB не активированы.
- Биты состояния STS\_DO и STS\_CMP (компаратор состояния в IDB) сброшены.

**Параметры, неназначаемые в блоке (статические локальные данные)**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>STS_CMP</b>	STATIC	BOOL	26.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние компаратора. Сбрасывается посредством RES_STS. Бит состояния STS_CMP показывает, что Условия для сравнения для компаратора выполнены или были выполнены. STS_CMP также индицирует, что выход был установлен: (STS_DO = TRUE (ИСТИНА))
<b>STS_OFLW</b>	STATIC	BOOL	26.5	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние переполнения счетчика. Сброс с помощью RES_STS.
<b>STS_UFLW</b>	STATIC	BOOL	26.6	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние потери значимости счетчика. Сброс с помощью RES_STS.
<b>STS_ZP</b>	STATIC	BOOL	26.7	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние перехода через ноль. Сброс с помощью RES_STS. Устанавливается только при отсутствии главного направления счета. Индицирует переход через 0. Параметр также установлен, когда счетчик установлен в 0, или при начале счета со значения 0.
<b>JOB_OVAL</b>	STATIC	DINT	28	$-2^{31}$ ... ( $+2^{31} - 1$ )	0	Выходное значение для заданий считывания.

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Сброс битов состояния. Выполняет сброс следующих битов состояния: STS_CMP, STS_OFLW, STS_UFLW и STS_ZP. Для сброса битов состояния требуются два вызова SFB.

## Информация об ошибках

### Ошибки при выполнении задания

Если при выполнении задания возникает ошибка, выходной параметр JOB\_ERR устанавливается в состояние TRUE (ИСТИНА). Параметр JOB\_STAT показывает причину ошибки.

Для получения информации по корректным значениям отдельных параметров обратитесь к руководству для пользователей.

Код ошибки	Объяснение
W#16#0121	Результат сравнения слишком мал.
W#16#0122	Результат сравнения слишком велик.
W#16#0131	Величина гистерезиса слишком мала.
W#16#0132	Величина гистерезиса слишком велика.
W#16#0141	Величина периода следования импульсов слишком мала.
W#16#0142	Величина периода следования импульсов слишком велика.
W#16#0151	Величина загруженного значения слишком мала.
W#16#0152	Величина загруженного значения слишком велика.
W#16#0161	Значение счетчика слишком мало.
W#16#0162	Значение счетчика слишком велико.
W#16#01FF	Некорректный номер задания.



### Системные ошибки

Если возникает системная ошибка, параметр BIE = FALSE (ЛОЖЬ)

Код ошибки	Объяснение
W#16#8001	Неправильный рабочий режим или неверно заданы параметры. Корректно задайте рабочий режим в "Configure Hardware" (конфигурация оборудования) или используйте специальный SFB для настройки рабочего режима.
W#16#8009	Неверный номер канала (CHANNEL). Установите номер канала не выше 3 (значение определяется типом CPU).

## 26.4 Управление измерением частоты с использованием SFB 48 "FREQUENCY"

### Описание

Чтобы управлять измерением частоты с помощью программы пользователя, используйте **SFB 48 "FREQUENCY"**.

Функция обеспечивает следующие операции:

- Запуск / остановка частотомера через программный шлюз **SW\_GATE**.
- Включение / управление выходом DO.
- Восстановление состояния битов **STS\_CMP**, **STS\_OFLW** и **STS\_UFLW**.
- Восстановление фактического значения частоты **MEAS\_VAL**.
- Задания чтения / записи во внутренних регистрах измерительного счетчика.

### Параметры:

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>LADDR</b>	INPUT	WORD	0	Определяется типом CPU	W#16#0300	I/O адрес Вашего submodule, определенный в "HW Config". Если I и O адреса не равны, то младший из них должен быть определен.
<b>CHANNEL</b>	INPUT	INT	2	CPU 312C: 0 ... 1 CPU 313C: 0 ... 2 CPU 314C: 0 ... 3	0	Номер канала
<b>SW_GATE</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Программный шлюз для запуска и остановки измерительного счетчика.
<b>MAN_DO</b>	INPUT	BOOL	4.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Разрешение ручного управления выходом
<b>SET_DO</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управление выходом

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>JOB_REQ</b>	INPUT	BOOL	4.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания (по переднему фронту сигнала).
<b>JOB_ID</b>	INPUT	WORD	6	W#16#0000 задание без функции W#16#0001 запись нижнего предела W#16#0001 запись верхнего предела W#16#0004 запись времени интегрирования W#16#0081 чтение нижнего предела W#16#0081 чтение верхнего предела W#16#0084 чтение времени интегрирования	W#16#0000	Номер задания
<b>JOB_VAL</b>	INPUT	DINT	8	$-2^{31}$ ... $(+2^{31} - 1)$	0	Значение для задания записи
<b>STS_GATE</b>	OUTPUT	BOOL	12.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние внутреннего шлюза
<b>STS_STRT</b>	OUTPUT	BOOL	12.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние программного шлюза (вход запуска)
<b>STS_DO</b>	OUTPUT	BOOL	12.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние выхода

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
STS_C_DN	OUTPUT	BOOL	12.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние обратного счета счетчика. Всегда отображает направление последнего счета. Значение STS_C_DN равно FALSE (ЛОЖЬ) после первого вызова SFB.
STS_C_UP	OUTPUT	BOOL	12.4	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние прямого счета счетчика. Всегда отображает направление последнего счета. Значение STS_C_UP равно TRUE (ИСТИНА) после первого вызова SFB.
MEAS_VAL	OUTPUT	DINT	14	$0 \dots (+2^{31} - 1)$	0	Фактическое значение частоты
COUNTVAL	OUTPUT	DINT	18	$-2^{31} \dots (+2^{31} - 1)$	0	Фактическое значение счетчика (счетчик запускается всякий раз, когда шлюз открыт на 0)
JOB_DONE	OUTPUT	BOOL	22.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Новое задание может стартовать
JOB_ERR	OUTPUT	BOOL	22.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Сбой задания
JOB_STAT	OUTPUT	WORD	24	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок задания

### Примечание

Если Вы установили параметр "Reaction of the output" (реакция выходного значения) в состоянии "No comparison" (не сравнивать), то справедливо следующее:

- Выход будет включен как обычный выход.
- Входные параметры MAN\_DO и SET\_DO блока SFB не активированы.
- Бит состояния STS\_DO сброшен.

**Параметры, неназначаемые в блоке (статические локальные данные)**

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>STS_CMP</b>	STATIC	BOOL	26.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние окончания измерения. Сбрасывается посредством RES_STS. Измеренное значение всякий раз обновляется по истечении временного периода. STS_CMP также индицирует окончание измерения.
<b>STS_OFLW</b>	STATIC	BOOL	26.5	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние переполнения измерительного счетчика. Сброс с помощью RES_STS.
<b>STS_UFLW</b>	STATIC	BOOL	26.6	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние потери значимости измерительного счетчика. Сброс с помощью RES_STS.
<b>JOB_OVAL</b>	STATIC	DINT	28	$-2^{31}$ ... $(+2^{31} - 1)$	0	Выходное значение для заданий считывания.
<b>RES_STS</b>	STATIC	BOOL	32.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Сброс битов состояния. Выполняет сброс следующих битов состояния: STS_CMP, STS_OFLW, STS_UFLW. Для сброса битов состояния требуются два вызова SFB.

## Информация об ошибках

### Ошибки при выполнении задания

Если при выполнении задания возникает ошибка, выходной параметр JOB\_ERR устанавливается в состояние TRUE (ИСТИНА).  
Параметр JOB\_STAT показывает причину ошибки.

Для получения информации по корректным значениям отдельных параметров обратитесь к руководству для пользователей.

Код ошибки	Объяснение
W#16#0221	Величина времени интегрирования слишком мала.
W#16#0222	Величина времени интегрирования слишком велика.
W#16#0231	Значение нижнего предела частоты слишком мало.
W#16#0232	Значение верхнего предела частоты слишком велико.
W#16#0241	<i>Значение верхнего предела частоты слишком мало.</i>
W#16#0242	<i>Значение верхнего предела частоты слишком велико.</i>
W#16#02FF	Некорректный номер задания.

### Системные ошибки

Если возникает системная ошибка, параметр BIE = FALSE (ЛОЖЬ)

Код ошибки	Объяснение
W#16#8001	Неправильный рабочий режим или неверно заданы параметры. Корректно задайте рабочий режим в "Configure Hardware" (конфигурация оборудования) или используйте специальный SFB для настройки рабочего режима.
W#16#8009	Неверный номер канала (CHANNEL). Установите номер канала не выше 3 (значение определяется типом CPU).

## 26.5 Управление ШИМ-модуляцией с использованием SFB 49 "PULSE"

### Описание

Чтобы управлять ШИМ-модуляцией с помощью программы пользователя, используйте **SFB 49 "PULSE"**.

Функция обеспечивает следующие операции:

- Запуск / остановка частотомера через программный шлюз **SW\_EN**.
- Включение / управление выходом DO.
- Восстановление состояния битов **STS\_EN**, **STS\_STRT** и **STS\_DO**.
- Ввод выходного значения.
- Задания чтения / записи в регистрах.

### Параметры:

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>LADDR</b>	INPUT	WORD	0	Определяется типом CPU	W#16#0300	I/O адрес Вашего submodule, определенный в "HW Config". Если I и O адреса не равны, то младший из них должен быть определен.
<b>CHANNEL</b>	INPUT	INT	2	CPU 312C: 0 ... 1 CPU 313C: 0 ... 2 CPU 314C: 0 ... 3	0	Номер канала
<b>SW_EN</b>	INPUT	BOOL	4.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Программный шлюз для разрешения и запрета выходного сигнала.
<b>MAN_DO</b>	INPUT	BOOL	4.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Разрешение ручного управления выходом
<b>SET_DO</b>	INPUT	BOOL	4.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управление выходом

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
<b>OUTP_VAL</b>	INPUT	INT	6.0	Импульсов в минуту: 0 ... 1000; в диапазоне S7 аналоговых значений: 0 ... 27 648	0	Выходное значение по умолчанию. При задании больших значений, чем граничные: 1000 или 27 648, CPU ограничит вводимое значение соответственно величинами 1000 или 27 648.
<b>JOB_REQ</b>	INPUT	BOOL	8.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Инициализация задания (по переднему фронту сигнала).
<b>JOB_ID</b>	INPUT	WORD	10	W#16#0000 задание без функции W#16#0001 запись периода времени W#16#0001 запись времени задержки включения W#16#0004 запись минимального периода следования импульсов W#16#0081 чтение периода времени W#16#0081 чтение времени задержки включения W#16#0084 чтение минимального периода следования импульсов	W#16#0000	Номер задания
<b>JOB_VAL</b>	INPUT	DINT	12	$-2^{31}$ ... $(+2^{31} - 1)$	0	Значение для задания записи
<b>STS_EN</b>	OUTPUT	BOOL	16.0	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние внутреннего шлюза



Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
STS_STRT	OUTPUT	BOOL	16.1	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние программного шлюза (вход запуска)
STS_DO	OUTPUT	BOOL	16.2	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Состояние выхода
JOB_DONE	OUTPUT	BOOL	16.3	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	TRUE (ИСТИНА)	Новое задание может стартовать
JOB_ERR	OUTPUT	BOOL	16.4	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Сбой задания
JOB_STAT	OUTPUT	WORD	18	W#16#0000 ... W#16#FFFF	W#16#0000	Число ошибок задания

#### Параметры, неназначаемые в блоке (статические локальные данные)

Параметр	Объявление	Тип данных	Адрес (экземпляр DB)	Диапазон значений	Значение по умолчанию	Описание
JOB_OVAL	STATIC	DINT	20	$-2^{31}$ ... $(+2^{31} - 1)$	0	Выходное значение для заданий считывания.

#### Информация об ошибках

##### Ошибки при выполнении задания

Если при выполнении задания возникает ошибка, выходной параметр JOB\_ERR устанавливается в состояние TRUE (ИСТИНА)). Параметр JOB\_STAT показывает причину ошибки.

Для получения информации по корректным значениям отдельных параметров обратитесь к руководству для пользователей.

Код ошибки	Объяснение
W#16#0411	Величина периода времени слишком мала.
W#16#0412	Величина периода времени слишком велика.
W#16#0421	Величина времени задержки включения слишком мала.
W#16#0422	Величина времени задержки включения слишком велика.

Код ошибки	Объяснение
W#16#0431	Значение минимального периода следования импульсов слишком мало.
W#16#0432	Значение минимального периода следования импульсов слишком велико.
W#16#04FF	Некорректный номер задания.

### Системные ошибки

Если возникает системная ошибка, параметр BIE = FALSE (ЛОЖЬ)

Код ошибки	Объяснение
W#16#8001	Неправильный рабочий режим или неверно заданы параметры. Корректно задайте рабочий режим в "Configure Hardware" (конфигурация оборудования) или используйте специальный SFB для настройки рабочего режима.
W#16#8009	Неверный номер канала (CHANNEL). Установите номер канала не выше 3 (значение определяется типом CPU).

## 26.6 Пересылка данных (ASCII, 3964(R)) с использованием SFB 60 "SEND\_PTP"

### Описание

Вы можете передавать блок данных из DB, используя **SFB 60 "SEND\_PTP"**.

Операция пересылки выполняется после вызова блока и после появления переднего фронта сигнала на управляющем входе **REQ**.

Диапазон данных для передачи определяется в **SD\_1** (номер DB и начальный адрес). Длина блока данных определяется в параметре **LEN**.

Для того, чтобы SFB начал выполнять задание по пересылке данных необходимо вызвать его с параметром **R** (Reset) = FALSE (ЛОЖЬ). При появлении переднего фронта сигнала на управляющем входе R текущая передача данных прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке (STATUS).

Для Вашего submodule Вы должны задать I/O адрес, который Вы можете определить с помощью утилиты "HW Config", в параметре **LADDR**.

Параметр **DONE** устанавливается в состояние TRUE (ИСТИНА), если задание завершается без ошибок. При возникновении ошибок при передаче параметр **ERROR** устанавливается в состояние TRUE (ИСТИНА).

Если задание завершается с параметром **DONE = TRUE** (ИСТИНА), то это означает, что:

- При использовании драйвера ASCII:  
Данные были переданы партнеру по связи. Но это не означает, что все данные были приняты партнером по связи.
- При использовании процедуры 3964(R):  
Данные были переданы партнеру по связи, получение данных было подтверждено партнером. Но это не означает, что данные были переданы CPU партнера по связи.

В параметре **STATUS** индицируется ошибка или соответствующий ID события, если при передаче было получено предупреждение.

DONE или ERROR/STATUS также реагируют, если SFB сбрасывается (R = TRUE (ИСТИНА)).

Если возникает системная ошибка, двоичный параметр BIE = FALSE (ЛОЖЬ). Если не возникает ошибок при обработке блока, BIE = TRUE (ИСТИНА).

**Примечание**

Проверка параметров не включена в SFB. CPU может перейти в режим STOP, если произойдет сбой из-за ошибок в параметрах.

**Экземпляр DB**

Блок SFB SEND\_PTP обрабатывается вместе с экземпляром DB. Номер DB передается вместе с вызовом. Доступ к данным в экземпляре DB не разрешен.

**Параметры:**

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>REQ</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Запрос". Инициализация задания (по переднему фронту сигнала).
<b>R</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Сброс". Прерывание задания. Передача блокируется.
<b>LADDR</b>	INPUT	WORD	Определяется типом CPU	W#16#03FF	I/O адрес Вашего submodule, определенный в "HW Config".
<b>DONE</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): <ul style="list-style-type: none"> <li>FALSE (ЛОЖЬ): задание выполняется или пока не стартовало</li> <li>TRUE (ИСТИНА): задание выполнено без ошибок.</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): Завершение задания без ошибок.

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 ... W#16#FFFF	W#16#0000	<p>Параметр состояния (устанавливается только на период одного вызова). Для отображения состояния Вы должны скопировать параметр STATUS в свободную область данных. Параметр имеет разные значения в зависимости от битов ERROR:</p> <ul style="list-style-type: none"> <li>• ERROR = FALSE (ЛОЖЬ): STATUS = W#16#0000: нет предупреждений или ошибок. STATUS ≠ W#16#0000: предупреждение - STATUS содержит детальную информацию.</li> <li>• ERROR = TRUE (ИСТИНА): Произошла ошибка. STATUS содержит детальную информацию.</li> </ul>
<b>SD_1:</b>	INPUT/ OUTPUT	ANY	Определяется типом CPU	0	<p>Параметры передачи: Здесь Вы должны указать следующие параметры:</p> <ul style="list-style-type: none"> <li>• номер DB, из которого должны пересылаться данные;</li> <li>• номер байта данных, начиная с которого должны пересылаться данные.</li> </ul> <p>Например: DB 10 с байта 2 -&gt; DB10.DBB2</p>
<b>LEN</b>	INPUT/ OUTPUT	INT	1 ... 1024	1	<p>Здесь задается длина блока данных, который должен быть передан. (Длина устанавливается косвенно).</p>

### **Консистентность данных**

Консистентные данные могут иметь размер максимум 206 байтов. Если Вам необходимо переслать как консистентные данные большой объем информации, Вы должны принимать в расчет следующее:

Не записывайте до окончания пересылки данных в используемый для пересылки раздел. Когда пересылка данных будет завершена, параметр DONE получит значение TRUE (ИСТИНА).

## 26.7 Прием данных (ASCII, 3964(R)) с использованием SFB 61 "RCV\_PTP"

### Описание

Вы можете принимать данные и, в дальнейшем, формировать из них блок данных, используя **SFB 61 "RCV\_PTP"**.

Операция пересылки выполняется после вызова блока и после установления на управляющем входе **EN\_R** состояния TRUE (ИСТИНА). Вы можете прекратить прием данных при сбросе управляющего входе **EN\_R** (**EN\_R** = FALSE (ЛОЖЬ)). Текущая передача данных при этом прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке в параметре **STATUS**.

Область памяти для приема данных определяется в параметре **RD\_1** (номер DB и начальный адрес). Длина блока данных определяется в параметре **LEN**.

Для того, чтобы SFB начал выполнять задание по пересылке данных необходимо вызвать его с параметром **R** (Reset) = FALSE (ЛОЖЬ). При появлении переднего фронта сигнала на управляющем входе **R** текущая передача данных прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке (**STATUS**).

Для Вашего submodule Вы должны задать I/O адрес, который Вы можете определить с помощью утилиты "HW Config", в параметре **LADDR**.

Параметр **NDR** устанавливается в состояние TRUE (ИСТИНА), если задание завершается без ошибок. При возникновении ошибок параметр **ERROR** устанавливается в состояние TRUE (ИСТИНА).

В параметре **STATUS** индицируется ошибка или соответствующий ID события, если было получено предупреждение.

**NDR** или **ERROR/STATUS** также реагируют, если SFB сбрасывается (**R** = TRUE (ИСТИНА)).

Если возникает системная ошибка, двоичный параметр **BIE** = FALSE (ЛОЖЬ). Если не возникает ошибок при обработке блока, **BIE** = TRUE (ИСТИНА).

**Примечание**

Проверка параметров не включена в SFB. CPU может перейти в режим STOP, если произойдет сбой из-за ошибок в параметрах.

**Экземпляр DB**

Блок SFB RCV\_PTP обрабатывается вместе с экземпляром DB. Номер DB передается вместе с вызовом. Доступ к данным в экземпляре DB не разрешен.

**Параметры:**

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>EN_R</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Разрешение на прием".
<b>R</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Сброс". Прерывание задания.
<b>LADDR</b>	INPUT	WORD	Определяется типом CPU	W#16#03FF	I/O адрес Вашего submodule, определенный в "HW Config".
<b>NDR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния "Задание выполнено без ошибок, данные приняты" <ul style="list-style-type: none"> <li>• FALSE (ЛОЖЬ): задание выполняется или пока не стартовало</li> <li>• TRUE (ИСТИНА): задание выполнено без ошибок.</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): Завершение задания без ошибок.



Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 ... W#16#FFFF	W#16#0000	<p>Параметр состояния (устанавливается только на период одного вызова). Для отображения состояния Вы должны скопировать параметр STATUS в свободную область данных. Параметр имеет разные значения в зависимости от битов ERROR:</p> <ul style="list-style-type: none"> <li>• ERROR = FALSE (ЛОЖЬ): STATUS = W#16#0000: нет предупреждений или ошибок. STATUS ≠ W#16#0000: предупреждение - STATUS содержит детальную информацию.</li> <li>• ERROR = TRUE (ИСТИНА): Произошла ошибка. STATUS содержит детальную информацию о типе обнаруженных ошибок.</li> </ul>
<b>RD_1:</b>	INPUT/ OUTPUT	ANY	Определяется типом CPU	0	<p>Параметры передачи: Здесь Вы должны указать следующие параметры:</p> <ul style="list-style-type: none"> <li>• номер DB, в котором принимаемые данные должны сохраняться;</li> <li>• номер байта данных, начиная с которого должны сохраняться данные.</li> </ul> <p>Например: DB 20 с байта 5 -&gt; DB20.DBB5</p>
<b>LEN</b>	INPUT/ OUTPUT	INT	1 ... 1024	1	Здесь задается длина массива данных, который должен быть принят. (Число байтов).

### **Консистентность данных**

Консистентные данные могут иметь размер максимум 206 байтов. Если Вам необходимо переслать как консистентные данные большой объем информации, Вы должны принимать в расчет следующее:

Не записывайте до окончания пересылки данных в используемый для пересылки раздел. Когда пересылка данных будет завершена, параметр NDR получит значение TRUE (ИСТИНА).

## 26.8 Очистка буфера приема (ASCII, 3964(R)) с использованием SFB 62 "RES\_RCVB"

### Описание

Вы можете целиком очистить входной буфер данных модуля, используя блок **SFB 62 "RES\_RCVB"**. Все посланные данные будут стерты. Фрейм входящих сообщений, тем не менее, остается при вызове SFB RES\_RCVB.

Операция очистки входного буфера выполняется после вызова блока при появлении на управляющем входе **REQ** переднего фронта сигнала. Задание может выполняться в течение нескольких программных циклов (перекрывать по времени несколько вызовов блока).

Для того, чтобы активировать работу SFB, Вы должны вызвать его с параметром **R** (Reset) = FALSE (ЛОЖЬ). При появлении на управляющем входе **R** положительного (переднего) фронта сигнала процесс очистки буфера прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке в параметре STATUS.

Для Вашего submodule Вы должны задать I/O адрес, который Вы можете определить с помощью утилиты "HW Config", в параметре **LADDR**.

Параметр **DONE** устанавливается в состояние TRUE (ИСТИНА), если задание завершается без ошибок. При возникновении ошибок параметр **ERROR** устанавливается в состояние TRUE (ИСТИНА).

В параметре **STATUS** индицируется ошибка или соответствующий ID события, если было получено предупреждение.

DONE или ERROR/STATUS также реагируют, если SFB сбрасывается (R = TRUE (ИСТИНА)).

Если возникает системная ошибка, двоичный параметр BIE = FALSE (ЛОЖЬ). Если не возникает ошибок при обработке блока, BIE = TRUE (ИСТИНА).

---

### Примечание

Проверка параметров не включена в SFB. CPU может перейти в режим STOP, если произойдет сбой из-за ошибок в параметрах.

---

### Экземпляр DB

Блок SFB RES\_RCVB обрабатывается вместе с экземпляром DB. Номер DB передается вместе с вызовом. Доступ к данным в экземпляре DB не разрешен.

### Параметры:

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>REQ</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Запрос". (По переднему фронту сигнала)
<b>R</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Сброс". Прерывание задания.
<b>LADDR</b>	INPUT	WORD	Определяется типом CPU	W#16#03FF	I/O адрес Вашего submodule, определенный в "HW Config".
<b>DONE</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния "Задание выполнено без ошибок" (Параметр устанавливается на период только одного вызова) <ul style="list-style-type: none"> <li>FALSE (ЛОЖЬ): задание выполняется или пока не стартовало</li> <li>TRUE (ИСТИНА): задание выполнено без ошибок.</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): Завершение задания без ошибок.
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 ... W#16#FFFF	W#16#0000	Параметр состояния (устанавливается только на период одного вызова). Для отображения состояния Вы должны скопировать параметр STATUS в свободную область данных. Параметр имеет разные значения в зависимости от битов ERROR: <ul style="list-style-type: none"> <li>ERROR = FALSE (ЛОЖЬ): STATUS = W#16#0000: нет предупреждений или ошибок. STATUS ≠ W#16#0000: предупреждение - STATUS содержит детальную информацию.</li> </ul>

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
					<ul style="list-style-type: none"><li>• ERROR = TRUE (ИСТИНА): Произошла ошибка. STATUS содержит детальную информацию о типе обнаруженных ошибок.</li></ul>

## 26.9 Пересылка данных (512(R)) с использованием SFB 63 "SEND\_RK"

### Описание

Вы можете переслать данные из блока данных, используя блок **SFB 63 "SEND\_RK"**.

Операция пересылки выполняется после вызова блока и после появления переднего фронта сигнала на управляющем входе **REQ**.

Диапазон данных для передачи определяется в **SD\_1** (номер DB и начальный адрес). Длина блока данных определяется в параметре **LEN**.

В SFB Вы также должны объявить область приема станции Вашего партнера по связи. CPU введет эту информацию в заголовок фрейма сообщения (См. приложение F) и перешлет его Вашему партнеру по связи.

Получатель определяется по номеру CPU **R\_CPU** (только для многопроцессорной связи), по типу данных в параметре **R\_TYPE** (блоки данных (DB) и блоки данных расширения (DX)), по номеру блока данных в параметре **R\_DBNO** и по смещению в параметре **R\_OFFSET**, с помощью которого определяется позиция первого байта.

В параметрах **R\_CF\_BYT** и **R\_CF\_BIT** объявляются байт и бит соединения в памяти CPU партнера по связи.

В параметре **SYNC\_DB** объявляется DB, в котором будут храниться данные, которые Вы используете во всех SFB для инициализации и синхронизации. Номер DB должен быть идентичен для всех SFB в Вашей программе пользователя.

Для того, чтобы SFB начал выполнять задание по пересылке данных необходимо вызвать его с параметром **R** (Reset) = FALSE (ЛОЖЬ). При появлении переднего фронта сигнала на управляющем входе R текущая передача данных прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке (STATUS).

Для Вашего submodule Вы должны задать I/O адрес, который Вы можете определить с помощью утилиты "HW Config", в параметре **LADDR**.

Параметр **DONE** устанавливается в состояние TRUE (ИСТИНА), если задание завершается без ошибок. При возникновении ошибок при передаче параметр **ERROR** устанавливается в состояние TRUE (ИСТИНА).

В параметре **STATUS** индицируется ошибка или соответствующий ID события, если при передаче было получено предупреждение.

DONE или ERROR/STATUS также реагируют, если SFB сбрасывается (R = TRUE (ИСТИНА)).

Если возникает системная ошибка, двоичный параметр BIE = FALSE (ЛОЖЬ).  
Если не возникает ошибок при обработке блока, BIE = TRUE (ИСТИНА).

---

#### Примечание

Проверка параметров не включена в SFB. CPU может перейти в режим STOP, если произойдет сбой из-за ошибок в параметрах.

---

#### Экземпляр DB

Блок SFB SEND\_RK обрабатывается вместе с экземпляром DB. Номер DB передается вместе с вызовом. Доступ к данным в экземпляре DB не разрешен.

#### Особенности при пересылке данных

Примите во внимание следующее при выполнении задания по пересылке с данным SFB:

- С помощью RK512 Вы можете посылать только четное число данных. Если Вы объявите нечетное значение длины (LEN), то дополнительный "пустой" байт со значением "0" будет добавлен к передаваемым данным.
- С помощью RK512 Вы можете объявить только четное значение для смещения. Если Вы объявите нечетное значение смещения для сохранения данных, то на станции партнера по связи данные будут сохраняться со следующего наименьшего четного смещения.

Пример:

Вы задали смещение 7.

Данные будут сохраняться с байта 6.

#### Параметры:

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
SYNC_DB	INPUT	INT	Определяется типом CPU (0 не допускается)	0	Номер DB, в котором хранятся общие данные для синхронизации RK-SFB (минимальная длина = 240 байтов)

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>REQ</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Запрос". Инициализация задания (по переднему фронту сигнала).
<b>R</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Сброс". Прерывание задания. Передача блокируется.
<b>LADDR</b>	INPUT	WORD	Определяется типом CPU	W#16#03FF	I/O адрес Вашего субмодуля, определенный в "HW Config".
<b>R_CPU</b>	INPUT	INT	0 ... 4	1	Номер CPU партнера (только для много-процессорной работы)
<b>R_TYPE</b>	INPUT	CHAR	'D', 'X'	"D"	Тип адреса в CPU партнера (символы только в верхнем регистре) 'D': Блок данных 'X': Блок данных расширения
<b>R_DBNO</b>	INPUT	INT	0 ... 255	0	Номер блока данных в CPU партнера по связи
<b>R_OFFSET</b>	INPUT	INT	0 ... 510 (только четные значения)	0	Номер байта данных в CPU партнера по связи
<b>R_CF_BYT</b>	INPUT	INT	0 ... 255	255	Номер байта памяти соединения в CPU партнера по связи (255 означает нет байтов памяти соединения)
<b>R_CF_BIT</b>	INPUT	INT	0 ... 7	0	Номер бита памяти соединения в CPU партнера по связи
<b>DONE</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): • FALSE (ЛОЖЬ): задание выполняется или пока не стартовало • TRUE (ИСТИНА): задание выполнено без ошибок.
<b>ERROR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): Завершение задания без ошибок.



Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 ... W#16#FFFF	W#16#0000	<p>Параметр состояния (устанавливается только на период одного вызова). Для отображения состояния Вы должны скопировать параметр STATUS в свободную область данных. Параметр имеет разные значения в зависимости от битов ERROR:</p> <ul style="list-style-type: none"> <li>• ERROR = FALSE (ЛОЖЬ): STATUS = W#16#0000: нет предупреждений или ошибок. STATUS ≠ W#16#0000: предупреждение - STATUS содержит детальную информацию.</li> <li>• ERROR = TRUE (ИСТИНА): Произошла ошибка. STATUS содержит детальную информацию.</li> </ul>
<b>SD_1:</b>	INPUT/ OUTPUT	ANY	Определяется типом CPU	0	<p>Параметры передачи: Здесь Вы должны указать следующие параметры:</p> <ul style="list-style-type: none"> <li>• номер DB, из которого должны пересылаться данные;</li> <li>• номер байта данных, начиная с которого должны пересылаться данные.</li> </ul> <p>Например: DB 10 с байта 2 -&gt; DB10.DBB2</p>
<b>LEN</b>	INPUT/ OUTPUT	INT	1 ... 1024	1	<p>Здесь задается длина блока данных, который должен быть передан. (Длина устанавливается косвенно).</p>

### Объявления в заголовке фрейма сообщения

Ниже представлена таблица, где показаны объявления в заголовке фрейма сообщения RK512:

Источник в Вашей S7 (локальный CPU)	Получатель, партнер по связи (CPU партнера)	Заголовок фрейма сообщения, в байтах		
		3/4 Instruction type (тип инструкции)	5/6 D-DBNR/D Offset (смещение)	7/8 Number in (внутренний номер)
Блок данных	Блок данных	AD	DB/DW	Word
Блок данных	Блок данных расширения	AD	DB/DW	Word

Ниже дана расшифровка использованных аббревиатур:

D-DBNR	Номер блока данных назначения
D Offset	Начальный адрес назначения
DW	Смещение в машинных словах

### Консистентность данных

Консистентные данные могут иметь размер максимум 128 байтов. Если Вам необходимо переслать как консистентные данные большой объем информации, Вы должны принимать в расчет следующее:

Не записывайте до окончания пересылки данных в используемый для пересылки раздел. Когда пересылка данных будет завершена, параметр DONE получит значение TRUE (ИСТИНА).

## 26.10 Выборка данных (512(R)) с использованием SFB 64 "FETCH RK"

### Описание

Вы можете выбрать блок данных из станции партнера по связи, используя блок **SFB 64 "FETCH\_RK"**.

Операция выполняется после вызова блока и после появления переднего фронта сигнала на управляющем входе **REQ**.

Диапазон данных для передачи определяется в **RD\_1** (номер DB и начальный адрес). Длина блока данных определяется в параметре **LEN**.

В SFB Вы также должны объявить область пересылки станции Вашего партнера по связи. CPU введет эту информацию в заголовок фрейма сообщения и перешлет его Вашему партнеру по связи.

Получатель определяется по номеру CPU **R\_CPU** (только для многопроцессорной связи), по типу данных в параметре **R\_TYPE** (блоки данных (DB) и блоки данных расширения (DX), меркеры, входы, выходы, счетчики и таймеры), по номеру блока данных в параметре **R\_DBNO** и по смещению в параметре **R\_OFFSET**, с помощью которого определяется позиция первого байта.

В параметрах **R\_CF\_BYT** и **R\_CF\_BIT** объявляются байт и бит соединения в памяти CPU партнера по связи.

В параметре **SYNC\_DB** объявляется DB, в котором будут храниться данные, которые Вы используете во всех SFB для инициализации и синхронизации. Номер DB должен быть идентичен для всех SFB в Вашей программе пользователя.

Для того, чтобы SFB начал выполнять задание по пересылке данных необходимо вызвать его с параметром **R (Reset) = FALSE** (ЛОЖЬ). При появлении переднего фронта сигнала на управляющем входе R текущая передача данных прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке (**STATUS**).

Для Вашего submodule Вы должны задать I/O адрес, который Вы можете определить с помощью утилиты "HW Config", в параметре **LADDR**.

Параметр **DONE** устанавливается в состояние TRUE (ИСТИНА), если задание завершается без ошибок. При возникновении ошибок при передаче параметр **ERROR** устанавливается в состояние TRUE (ИСТИНА).

В параметре **STATUS** индицируется ошибка или соответствующий ID события, если при передаче было получено предупреждение.

DONE или ERROR/STATUS также реагируют, если SFB сбрасывается (**R = TRUE** (ИСТИНА)).

Если возникает системная ошибка, двоичный параметр BIE = FALSE (ЛОЖЬ).  
Если не возникает ошибок при обработке блока, BIE = TRUE (ИСТИНА).

---

**Примечание**

Проверка параметров не включена в SFB. CPU может перейти в режим STOP, если произойдет сбой из-за ошибок в параметрах.

---

**Примечание**

После выборки данных, Вы должны запрограммировать SFB "SERVE\_RK" для Вашего CPU.

---

**Экземпляр DB**

Блок SFB FETCH\_RK обрабатывается вместе с экземпляром DB. Номер DB передается вместе с вызовом. Доступ к данным в экземпляре DB не разрешен.

**Особенности применения при пересылке данных**

Примите во внимание следующее при выполнении задания по получению данных:

- С помощью RK512 Вы можете посылать только четное число данных. Если Вы объявите нечетное значение длины (LEN), то дополнительный "пустой" байт со значением "0" будет добавлен к передаваемым данным.
- С помощью RK512 Вы можете объявить только четное значение для смещения. Если Вы объявите нечетное значение смещения для сохранения данных, то на станции партнера по связи данные будут сохраняться со следующего наименьшего четного смещения.

Пример:

Вы задали смещение 7.

Данные будут сохраняться, начиная с байта 6.

**Особенности применения для таймеров и счетчиков**

При выборке данных от таймеров и счетчиков станции Вашего партнера по связи Вы должны учитывать необходимость выборки по два байта от каждого таймера и счетчика. Например, если Вы делаете выборку данных от 10 счетчиков, Вы должны объявить длину массива выбираемых данных, равную 20 байтов.

**Параметры:**

Примечание: В данном SFB принят германский вариант обозначения полей памяти.

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>SYNC_DB</b>	INPUT	INT	Определяется типом CPU (0 не допускается)	0	Номер DB, в котором хранятся общие данные для синхронизации RK-SFB (минимальная длина = 240 байтов)
<b>REQ</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Запрос". Инициализация задания (по переднему фронту сигнала).
<b>R</b>	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Сброс". Прерывание задания. Передача блокируется.
<b>LADDR</b>	INPUT	WORD	Определяется типом CPU	W#16#03FF	I/O адрес Вашего submodule, определенный в "HW Config".
<b>R_CPU</b>	INPUT	INT	0 ... 4	1	Номер CPU партнера (только для много-процессорной работы)
<b>R_TYPE</b>	INPUT	CHAR	'D', 'X', 'M', 'E', 'A', 'Z', 'T'	"D"	Тип адреса в CPU партнера (символы только в верхнем регистре) 'D': блок данных 'X': расширенный блок данных 'M': меркер 'E': вход 'A': выход 'Z': счетчик 'T': таймер
<b>R_DBNO</b>	INPUT	INT	0 ... 255	0	Номер блока данных в CPU партнера по связи
<b>R_OFFSET</b>	INPUT	INT	См. таблицу "Параметры FB для источников данных"	0	Номер байта данных в CPU партнера по связи
<b>R_CF_BYT</b>	INPUT	INT	0 ... 255	255	Номер байта памяти соединения в CPU партнера по связи (255 означает нет байтов памяти соединения)
<b>R_CF_BIT</b>	INPUT	INT	0 ... 7	0	Номер бита памяти соединения в CPU партнера по связи

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>DONE</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): <ul style="list-style-type: none"> <li>FALSE (ЛОЖЬ): задание выполняется или пока не стартовало</li> <li>TRUE (ИСТИНА): задание выполнено без ошибок.</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): Завершение задания без ошибок.
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 ... W#16#FFFF	W#16#0000	Параметр состояния (устанавливается только на период одного вызова). Для отображения состояния Вы должны скопировать параметр STATUS в свободную область данных. Параметр имеет разные значения в зависимости от битов ERROR: <ul style="list-style-type: none"> <li>ERROR = FALSE (ЛОЖЬ): STATUS = W#16#0000: нет предупреждений или ошибок. STATUS ≠ W#16#0000: предупреждение - STATUS содержит детальную информацию.</li> <li>ERROR = TRUE (ИСТИНА): Произошла ошибка. STATUS содержит детальную информацию.</li> </ul>
<b>RD_1:</b>	INPUT/ OUTPUT	ANY	Определяется типом CPU	0	Параметры приема: Здесь Вы должны указать следующие параметры: <ul style="list-style-type: none"> <li>номер DB, в котором будут сохранены данные;</li> <li>номер байта данных, начиная с которого должны сохраняться данные.</li> </ul> Например: DB 10 с байта 2 -> DB10.DBB2

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
LEN	INPUT/ OUTPUT	INT	1 ... 1024	1	Здесь задается длина блока данных, который должен быть выбран. (Длина устанавливается косвенно).

### Параметры в SFB для источника данных (CPU партнера по связи)

Ниже представлена таблица, где показаны типы данных для передачи. Значение для R\_OFFSET определяется партнером по связи:

Источник в партнер по связи (CPU партнера)	R_TYP	R_NO	R_OFFSET (в байтах)
Блок данных	'D'	0-255	0-510 (только четные значения)
Блок данных расширения	'X'	0-255	0-510 (только четные значения)
Меркер	'M'	-	0-255
Входы	'E'	-	0-255
Выходы	'A'	-	0-255
Счетчики	'Z'	-	0-255
Таймеры	'T'	-	0-255

### Объявления в заголовке фрейма сообщения

Ниже представлена таблица, где показаны объявления в заголовке фрейма сообщения RK512:

Источник в партнер по связи (CPU партнера)	Получатель Вашей S7 (локальный CPU)	Заголовок фрейма сообщения, в байтах		
		3/4 Instruction type (тип инструкции)	5/6 S-DBNR/S Offset (смещение)	7/8 Number in (внутренний номер)
Блок данных	Блок данных	ED	DB/DW	Word
Блок данных расширения	Блок данных	EX	DB/DW	Word
Меркер	Блок данных	EM	Адрес байта	Byte
Входы	Блок данных	EI	Адрес байта	Byte
Выходы	Блок данных	EO	Адрес байта	Byte
Счетчики	Блок данных	EC	Номер счетчика	Word
Таймеры	Блок данных	ET	Номер таймера	Word

Ниже дана расшифровка использованных аббревиатур:

D-DBNO	Номер блока данных источника
S Offset	Начальный адрес источника

### **Консистентность данных**

Консистентные данные могут иметь размер максимум 128 байтов. Если Вам необходимо переслать как консистентные данные большой объем информации, Вы должны принимать в расчет следующее:

Не записывайте до окончания пересылки данных в используемый для пересылки раздел. Когда пересылка данных будет завершена, параметр DONE получит значение TRUE (ИСТИНА).



## 26.11 Прием и выдача данных (512(R)) с использованием SFB 65 "SERVE\_RK"

### Описание

Для того, чтобы получать и выдавать данные используйте **SFB 65 "SERVE\_RK"**.

- Получение данных: данные сохраняются в области памяти, которая определена партнером в заголовке фрейма сообщения RK512. Вызов данного блока SFB требуется, когда партнер связи выполняет задание "передача данных" (SEND).
- Выдача данных: данные выбираются из области, которая определена партнером в заголовке фрейма сообщения RK512. Вызов данного блока SFB требуется, когда партнер связи выполняет задание "выборка данных" (FETCH).

Блок готов для обработки после того, как он будет вызван с управляющим входным параметром **EN\_R = TRUE (ИСТИНА)**. Вы можете отменить текущую передачу установкой параметра в состояние **FALSE (ЛОЖЬ)**. Отмененная работа завершается с сообщением об ошибках (STATUS). Прием данных невозможен, пока состояние сигнала параметра **EN\_R** не изменится на **TRUE (ИСТИНА)**.

В параметре **SYNC\_DB** Вы объявляете DB, в которых хранятся данные, используемые Вами во всех SFB для инициализации во время запуска и синхронизации.

Номера DB должны быть идентичны для всех SFB в Вашей программе пользователя.

Для того, чтобы SFB начал выполнять задание по обработке данных необходимо вызвать его с параметром **R (Reset) = FALSE (ЛОЖЬ)**. При появлении переднего фронта сигнала на управляющем входе **R** текущая передача данных прерывается и SFB переходит в исходное состояние. Прерванное задание завершается с сообщением об ошибке (STATUS).

Для Вашего submodule Вы должны задать I/O адрес, который Вы можете определить с помощью утилиты "HW Config", в параметре **LADDR**.

Параметр **NDR** устанавливается в состояние **TRUE (ИСТИНА)**, если задание завершается без ошибок. При возникновении ошибок параметр **ERROR** устанавливается в состояние **TRUE (ИСТИНА)**.

Если параметр **NDR** установлен в состояние **TRUE (ИСТИНА)** при вызове SFB, то в параметрах **L\_TYPE**, **L\_DBNO** и **L\_OFFSET** будет указана область памяти в которой, данные будут сохранены или из которой данные будут выбраны. Соответствующую выполняемому заданию информацию при вызове SFB несут параметры **L\_CF\_BYT** и **L\_CF\_BIT** и **LEN**.

В параметре **STATUS** индицируется ошибка или соответствующий ID события, если было получено предупреждение.

NDR или ERROR/STATUS также реагируют, если SFB сбрасывается (R = TRUE (ИСТИНА)).

Если возникает системная ошибка, двоичный параметр BIE = FALSE (ЛОЖЬ). Если не возникает ошибок при обработке блока, BIE = TRUE (ИСТИНА).

### Примечание

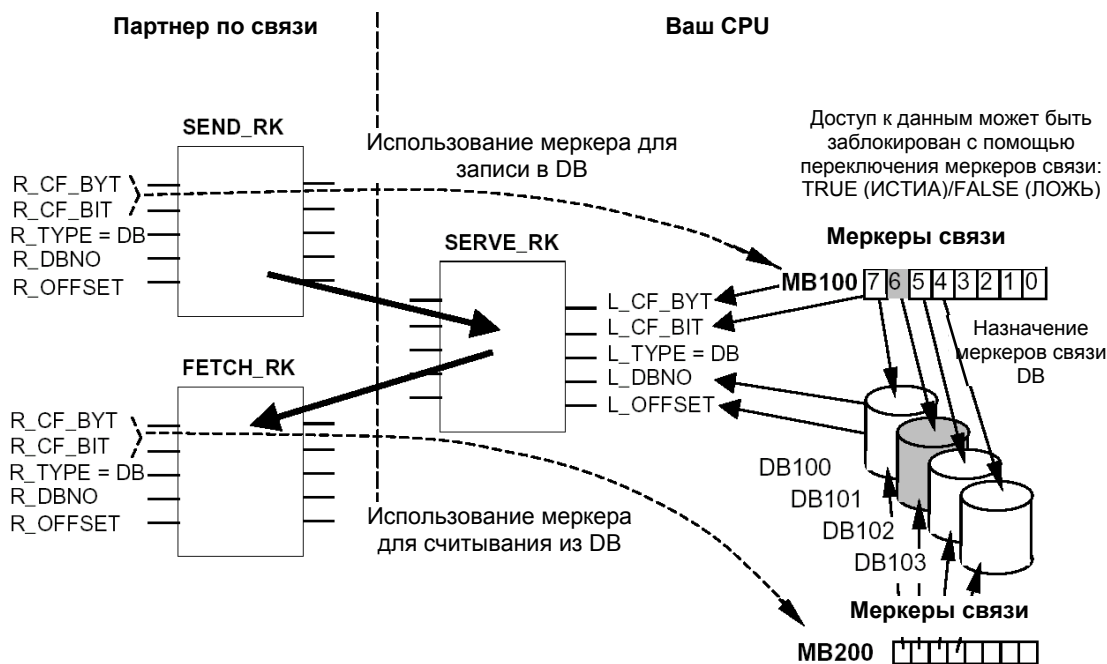
Проверка параметров не включена в SFB. CPU может перейти в режим STOP, если произойдет сбой из-за ошибок в параметрах.

## Экземпляр DB

Блок SFB RCV\_PTP обрабатывается вместе с экземпляром DB. Номер DB передается вместе с вызовом. Доступ к данным в экземпляре DB не разрешен.

## Как используются меркеры связи

Вы можете заблокировать или разрешить операции передачи (SEND) и выборки (FETCH) данных, выполняемые Вашим партнером по связи. Для этого используются меркеры связи. Таким образом Вы можете предотвращать запись поверх (замену) данных, которые еще Вами не обработаны. Вы можете назначать меркеры связи для каждого задания – см. рисунок ниже:



**Пример: SEND\_RK с меркером связи**

В этом примере партнер связи передает данные в DB 101 на Вашем CPU.

1. В Вашем CPU, установите бит 100.6 (меркер связи) в FALSE (ЛОЖЬ).
2. В Вашем партнере связи, определите бит 100.6 (параметры R\_CF\_BYT, R\_CF\_BIT) для выполнения задания пересылки (SEND). Меркер связи зафиксированн в Вашем CPU в RK 512 заголовка фрейма сообщения.

До начала выполнения задания CPU проверяет меркер связи, который определен в RK512 заголовке фрейма сообщения. Задание будет выполняться, только если меркер связи имеет значение FALSE (ЛОЖЬ) в Вашем CPU. Если меркер связи имеет значение TRUE (ИСТИНА), сообщение об ошибке "32H" передается в фрейме ответного сообщения партнеру связи.

После того, как данные пересылаются в DB101, SFB SERVE.установит в меркере связи 100.6 значение TRUE (ИСТИНА) в Вашем CPU. Байт для соединения и бит выводят информацию на SFB SERVE на протяжении одного вызова (если NDR = TRUE (ИСТИНА)).

3. Проверая меркер связи (меркер 100.6 = TRUE (ИСТИНА)) в Вашей программе пользователя, Вы можете определить, завершена ли работа, и если да, то переданные данные могут быть обработаны.
4. После того, как Вы обработали данные в Вашей программе пользователя, Вы должны сбросить меркер 100.6 к уровню FALSE (ЛОЖЬ). Только в этом случае Ваш партнер сможет снова выполнить работу без ошибки.

**Параметры**

Примечание: В данном SFB принят германский вариант обозначения полей памяти.

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
SYNC_DB	INPUT	INT	Определяется типом CPU	0	Номер DB, в котором сохраняются общие данные для синхронизации RK-SFB (минимальная длина = 240 байтов).
EN_R	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Разрешение на прием".
R	INPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Управляющий параметр "Сброс". Прерывание задания.

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>LADDR</b>	INPUT	WORD	Определяется типом CPU	W#16#03FF	I/O адрес Вашего submodule, определенный в "HW Config".
<b>NDR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния "Задание выполнено без ошибок, данные приняты" <ul style="list-style-type: none"> <li>FALSE (ЛОЖЬ): задание выполняется или пока не стартовало</li> <li>TRUE (ИСТИНА): задание выполнено без ошибок.</li> </ul>
<b>ERROR</b>	OUTPUT	BOOL	TRUE/FALSE (ИСТИНА/ЛОЖЬ)	FALSE (ЛОЖЬ)	Параметр состояния (устанавливается только на период одного вызова): Завершение задания без ошибок.
<b>STATUS</b>	OUTPUT	WORD	W#16#0000 ... W#16#FFFF	W#16#0000	Параметр состояния (устанавливается только на период одного вызова). Для отображения состояния Вы должны скопировать параметр STATUS в свободную область данных. Параметр имеет разные значения в зависимости от битов ERROR: <ul style="list-style-type: none"> <li>ERROR = FALSE (ЛОЖЬ): STATUS = W#16#0000: нет предупреждений или ошибок. STATUS ≠ W#16#0000: предупреждение - STATUS содержит детальную информацию.</li> <li>ERROR = TRUE (ИСТИНА): Произошла ошибка. STATUS содержит детальную информацию о типах обнаруженных ошибок.</li> </ul>

Параметр	Объявление	Тип данных	Диапазон значений	Значение по умолчанию	Описание
<b>L_TYPE</b>	OUTPUT	CHAR	'D', 'X', 'M', 'E', 'A', 'Z', 'T'	' '	Принимаемые данные: Тип адреса назначения в локального CPU (символы только в верхнем регистре) 'D': блок данных; Выдаваемые данные: 'D': блок данных 'M': меркер 'E': входы 'A': выходы 'Z': счетчики 'T': таймеры (Параметр устанавливается на протяжении одного вызова).
<b>L_DBNO</b>	OUTPUT	INT	Определяется типом CPU	0	Номер блока данных в локальном CPU. (Параметр устанавливается на протяжении одного вызова).
<b>L_OFFSET</b>	OUTPUT	INT	0 ... 510 (только четные значения)	0	Номер байта данных в локальном CPU. (Параметр устанавливается на протяжении одного вызова).
<b>L_CF_BYT</b>	OUTPUT	INT	0 ... 255	0	Номер байта меркеров в локальном CPU. (255 означает нет байтов меркеров)
<b>L_CF_BIT</b>	OUTPUT	INT	0 ... 7	0	Номер бита памяти соединения в локальном CPU.
<b>LEN</b>	INPUT/ OUTPUT	INT	1 ... 1024	0	Здесь задается длина фрейма сообщений. (Число байтов). (Параметр устанавливается на протяжении одного вызова).

### **Консистентность данных**

Консистентные данные могут иметь размер максимум 128 байтов. Если Вам необходимо переслать как консистентные данные большой объем информации, Вы должны принимать в расчет следующее:

Используйте функцию меркеров связи. Не разрешайте допуск к данным до полного окончания пересылки данных в используемый для пересылки раздел. (для этого используйте проверку состояния соответствующих меркеров; меркеры связи активны на протяжении одного вызова SFB, пока NDR = TRUE (ИСТИНА)). Не сбрасывайте меркеры связи в состояние FALSE (ЛОЖЬ), пока Вы не обработаете все данные.

## 26.12 Дополнительная информация об ошибках для SFB 60...65

### Информация об ошибках

В таблице, приведенной ниже, представлены различные классы событий и ID событий:

Ошибки при конфигурировании параметров SFB		
Код ошибки	Событие	Действия
W#16#0301	Некорректный или несуществующий тип исходных данных / данных назначения. Некорректно задана область памяти (начальный адрес, длина). Некорректный или несуществующий DB. Другие некорректные или несуществующие типы данных. Недопустимый ID байта меркеров (меркера) связи.	Проверьте правильность параметров, и исправьте их, если требуется. Партнер по связи передает запрещенные параметры в заголовке фрейма сообщения. Проверьте правильность параметров, создайте блок если требуется. Проверьте в таблицах задания, разрешены ли указанные типы данных. Партнер по связи передает неверные параметры в заголовке фрейма сообщения.
W#16#0303	Область памяти недоступна.	Проверьте правильность параметров. Проверьте в таблицах задания, разрешены ли начальные адреса и длины массивов данных. Проверьте правильность заданных партнером по связи параметров в заголовке фрейма сообщения.

Ошибки при выполнении задания		
Код ошибки	Событие	Действия
W#16#0501	Текущее задание было отменено при перезапуске или выполнении сброса (RESET).	Повторно запустите задание. При переназначении параметров с помощью программатора перед записью их в интерфейс убедитесь, что нет других выполняемых заданий.
W#16#0502	Выполнение задания не разрешено в данном рабочем режиме (например, не назначены параметры для интерфейса устройства).	Назначьте параметры для интерфейса устройства.
W#16#050E	<ul style="list-style-type: none"> <li>Неверно задана длина фрейма сообщения.</li> <li>Идентификатор конца сообщения, назначенный в параметрах, не обнаружен в максимально допустимом диапазоне.</li> </ul>	<ul style="list-style-type: none"> <li>Длина фрейма сообщения превышает 1024 бита. Выберите меньшую длину фрейма сообщения.</li> <li>Добавьте ID конца сообщения в требуемую позицию в буфере для передачи.</li> </ul>
W#16#0513	Ошибка типа данных (DB ...): Неизвестный тип данных или неверный тип данных (например, DE). Неадекватность типа исходных данных и типа данных назначения, определенных в SFB.	Проверьте в таблицах задания, разрешены ли указанные типы данных и их комбинации.
W#16#0515	Объявлен неправильный номер бита в области меркеров связи.	Разрешенные номера битов: 0 ... 7.
W#16#0516	Задан слишком большой номер CPU.	Разрешенные номера для CPU: 0, 1, 2, 3, 4.
W#16#0517	Передача данных размером более 1024 байтов запрещена.	Разделите большое задание на несколько мелких по размеру передаваемых данных.
W#16#051D	Передача / прием данных прерван по причине: <ul style="list-style-type: none"> <li>сброса (RESET) коммуникационного блока;</li> <li>переназначения параметров.</li> </ul>	Повторите вызов коммуникационного блока.
W#16#0522	Новое задание SEND (передача) запущено, хотя предыдущее задание еще не было завершено.	Не запускайте новых заданий SEND (передача) до завершения предыдущего задания из-за выполнения (DONE = TRUE (ИСТИНА)) или из-за ошибки (ERROR = TRUE (ИСТИНА)).



<b>Ошибки при обработке задания партнера по связи (только для RK512)</b>		
<b>Код ошибки</b>	<b>Событие</b>	<b>Действия</b>
W#16#0601	Ошибка в 1 байте инструкции (не 00 или FFH).	Ошибка основного заголовка структуры партнера по связи. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0602	Ошибка в 3 байте инструкции (не A, 0 или E).	Ошибка основного заголовка структуры партнера по связи. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0603	Ошибка в 3 байте инструкции в последовательности фреймов сообщений (инструкция не такая, как в 1 фрейме сообщения).	Ошибка основного заголовка структуры партнера по связи. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0604	Ошибка в 4 байте инструкции (ошибка в символе инструкции).	Ошибка основного заголовка структуры партнера по связи. Проверьте корректность инструкции. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0606	Ошибка в 5 байте инструкции (не правильный номер DB).	Проверьте в таблицах задания, разрешены ли указанные номера DB, начальные адреса и размеры (длины).
W#16#0607	Ошибка в 5 или 6 байте инструкции (начальный адрес слишком большой).	Проверьте в таблицах задания, разрешены ли указанные номера DB, начальные адреса и размеры (длины).
W#16#0609	Ошибка в 9 или 10 байте инструкции (меркер связи не разрешен для данного типа данных или номер его слишком велик).	Ошибка основного заголовка структуры партнера по связи. Проверьте в таблицах задания, разрешены ли меркеры связи.
W#16#060A	Ошибка в 10 байте инструкции (не правильный номер CPU).	Ошибка основного заголовка структуры партнера по связи.

Ошибки при выполнении задания SEND (пересылки)		
Код ошибки	Событие	Действия
W#16#0701	<p><b>Только для 3964(R):</b> Посылка первого повторения:</p> <ul style="list-style-type: none"> <li>• Ошибка была обнаружена при посылке фрейма сообщения</li> <li>• партнер запросил повторение с отрицательным символом подтверждения (NCC).</li> </ul>	<p>Повторение не означает ошибку. Тем не менее, оно может быть отображением факта сбоя на канале связи или сбойного поведения партнера. Если фрейм сообщения не удалось передать за установленное максимально возможное число повторений, то сообщается номер ошибки, который идентифицирует ошибку, произошедшую первой.</p>
W#16#0702	<p><b>Только для 3964(R):</b> Выявлена ошибка при установлении соединения. После передачи STX был получен сигнал NCC или любой другой символ (за исключением DLE или STX).</p>	<p>Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0703	<p><b>Только для 3964(R):</b> Превышено время ожидания подтверждения (QVZ): После передачи STX партнер не успел ответить за время задержки, отведенного для подтверждения.</p>	<p>Устройство партнера работает слишком медленно, или не готово получать информацию, или канал связи нарушен. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0704	<p><b>Только для 3964(R):</b> Произошло прерывание задания партнером: Один или несколько символов были получены от партнера по связи, в то время как задание по пересылке было активно.</p>	<p>Проверьте, индицирует ли партнер по связи также ошибку, потому что, возможно, он не получил все передаваемые данные (например, благодаря разорванному каналу связи), или из-за того, что фатальные ошибки возникли и ожидают реакции или поведение устройства партнера по связи некорректно. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0705	<p><b>Только для 3964(R):</b> Пришло отрицательное подтверждение, во время передачи информации.</p>	<p>Проверьте, индицирует ли партнер по связи также ошибку, потому что, возможно, он не получил все передаваемые данные (например, благодаря разорванному каналу связи), или из-за того, что фатальные ошибки возникли и ожидают реакции или поведение устройства партнера по связи некорректно. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>

Ошибки при выполнении задания SEND (пересылки)		
Код ошибки	Событие	Действия
W#16#0706	<p><b>Только для 3964(R):</b> Ошибка при окончании передачи::</p> <ul style="list-style-type: none"> <li>• партнер отверг фрейм сообщения в конце передачи с NCC или любым другим символом (за исключением DLE );</li> <li>• символ подтверждения (DLE) был получен слишком рано</li> </ul>	<p>Проверьте, индицирует ли партнер по связи также ошибку, потому что, возможно, он не получил все передаваемые данные (например, благодаря разорванному каналу связи), или из-за того, что фатальные ошибки возникли и ожидают реакции или поведение устройства партнера по связи некорректно.</p> <p>Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0707	<p><b>Только для 3964(R):</b> Время задержки для подтверждения окончания передачи / ответа с заданной задержкой (watchdog) было превышено: Партнер не ответил в течение интервала QVZ после того, как соединение было разорвано с помощью DLE ETX.</p>	<p>Устройство партнера работает слишком медленно или повреждено. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0708	<p><b>Только для ASCII drivers:</b> Время ожидания на XON истекло.</p>	<p>Партнер по связи отключен, слишком медленно работает или включен автономно.</p> <p>Проверьте параметры партнера по связи или измените параметры (если это требуется).</p>
W#16#0709	<p><b>Только для 3964(R):</b> Невозможно установить соединение, так как уже превышено максимально допустимое число попыток выполнения запуска.</p>	<p>Проверьте интерфейсный кабель, проверьте параметры передачи. Также необходимо проверить в партнере по связи правильность назначения параметров связи для функции передачи между CPU и CP.</p>
W#16#070A	<p><b>Только для 3964(R):</b> Передача данных невозможна, так как уже превышено максимально допустимое число попыток выполнения задания по передаче данных.</p>	<p>Проверьте интерфейсный кабель, проверьте параметры передачи.</p>
W#16#070B	<p><b>Только для 3964(R):</b> Конфликт инициализации не может быть разрешен из-за того, что для обоих параметров задан низкий приоритет.</p>	<p>Измените параметры.</p>
W#16#070C	<p><b>Только для 3964(R):</b> Конфликт инициализации не может быть разрешен из-за того, что для обоих параметров задан низкий приоритет.</p>	<p>Измените параметры.</p>

Ошибки при выполнении задания RECEIVE (прием)		
Код ошибки	Событие	Действия
W#16#0801	<p><b>Только для 3964(R):</b> Ожидание первого повторения: Ошибка была обнаружена, когда был получен фрейм сообщения, и CPU запросил повторение с отрицательным подтверждением (NCC) от партнера по связи.</p>	<p>Повторение не означает ошибку. Тем не менее, оно может быть отображением факта сбоя на канале связи или сбойного поведения партнера. Если фрейм сообщения не удалось передать за установленное максимально возможное число повторений, то сообщается номер ошибки, который идентифицирует ошибку, произошедшую первой.</p>
W#16#0802	<p><b>Только для 3964(R):</b> Ошибка при установлении соединения: • Один или несколько символов (за исключением NCC или STX) были получены во время простоя. • Получив STX, партнер по связи передал много символов, не ожидая ответа DLE.</p> <p>После включения устройства-партнера по связи: • CPU получает неопределенный символ, в то время как партнер включен.</p>	<p>Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0805	<p><b>Только для 3964(R):</b> Логическая ошибка: После получения DLE другой символ был получен (кроме DLE, ETX).</p>	<p>Проверьте, дублирует ли партнер DLE в заголовке фрейма сообщения и в строке данных или соединение установлено посредством DLE ETX.</p> <p>Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0806	<p>Время задержки символа (CDT) было превышено: • Два последующих символа не были получены на протяжении CDT.</p> <p><b>Только для 3964(R):</b> • Символ не был получен на протяжении CDT после посылки DLE, когда соединение было установлено.</p>	<p>Устройство партнера работает слишком медленно, или не готово получать информацию, или канал связи нарушен.</p> <p>Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.</p>
W#16#0807	<p><b>Только для 3964(R):</b> Неправильная длина фрейма сообщения: Был принят фрейм сообщения с нулевой длиной.</p>	<p>Получение фрейма сообщения с нулевой длиной не объясняет ошибку. Проверьте почему партнер по связи передает сообщения без данных пользователя.</p>

Ошибки при выполнении задания RECEIVE (прием)		
Код ошибки	Событие	Действия
W#16#0808	<b>Только для 3964(R):</b> Ошибка контрольного символа блока ВСС: Внутренне сгенерированное значение для ВСС не соответствует ВСС, полученному партнером в конце связи.	Проверьте наличие серьезного нарушения связи. При этом Вы можете также наблюдать коды случайных ошибок. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0809	<b>Только для 3964(R):</b> Время задержки для повторения блока истекло.	Вы должны объявлять одинаковое значение времени задержки для повторения блока в параметрах для партнера связи и для Вашего модуля. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#080A	Нет свободного входного буфера: Нет очищенного входного буфера, доступного для приема данных.	SFB RCV должен вызваться более часто.
W#16#080C	Ошибка передачи • Ошибка передачи была обнаружена (ошибка четности / ошибка стоп-бита / ошибка переполнения). <b>Только для 3964(R):</b> • Если символ прерывания будет получен во время неактивного режима, то немедленно будет сгенерировано сообщение об ошибке для своевременного и оперативного распознавания нарушения, возникшего в канале связи. <b>Только для 3964(R):</b> • Повторения выполняются, если активны режимы передачи / приема.	Помехи при обмене данными вызывают повторения фрейма сообщения и тем самым уменьшают скорость передачи данных пользователя. При этом растет риск не обнаружения ошибки. Измените структуру Вашей системы или измените монтаж. Проверьте каналы связи партнеров, а также одинаковость установки скорости обмена данными, контроля по четности и стоп-битов на обоих устройствах.
W#16#080D	Прерывание (BREAK): Произошло прерывание канала приема данных от партнера по связи.	Установите связь снова или включите устройство-партнера.
W#16#080E	Переполнение входного буфера в то время, когда управление потоком данных не разрешено.	SFB для получения данных должен вызваться более часто в программе пользователя или Вы должны назначить параметры для управления потоком данных в канале связи.
W#16#0810	Ошибка четности.	Проверьте каналы связи партнеров, а также одинаковость установки скорости обмена данными, контроля по четности и стоп-битов на обоих устройствах.

Ошибки при выполнении задания RECEIVE (прием)		
Код ошибки	Событие	Действия
W#16#0811	Ошибка символьного фрейма.	Проверьте каналы связи партнеров, а также одинаковость установки скорости обмена данными, контроля по четности и стоп-битов на обоих устройствах. Проверьте структуру Вашей системы или монтажных соединений.
W#16#0812	<b>Только для драйверов ASCII:</b> Были приняты последующие символы после того, как CPU передал XOFF.	Переназначьте параметры партнера по связи или увеличьте скорость обработки данных.
W#16#0814	<b>Только для драйверов ASCII:</b> Один или несколько фреймов сообщений были потеряны, потому что операция была проведена без управления потоком данных.	Максимально ускорьте управление потоком данных. Используйте весь входной буфер. В основных параметрах установите параметр "Reaction to CPU STOP" ("реакция на перевод CPU в STOP-режим") в состояние "Continue operation" ("продолжение работы").
W#16#0816	Длина полученного фрейма сообщения превысила максимально возможный размер.	Скорректируйте параметры в станции партнера по связи.

<b>"Получение ответного фрейма сообщения с ошибкой или фрейма сообщения об ошибках от партнера по связи"</b>		
<b>Код ошибки</b>	<b>Событие</b>	<b>Действия</b>
W#16#0902	<p><b>Только для RK 512:</b> Ошибка доступа к памяти в станции партнера по связи (указанная область памяти не существует). Для SIMATIC S5 как партнера по связи:</p> <ul style="list-style-type: none"> <li>• Ошибочный диапазон в слове отображения.</li> <li>• Диапазон данных не существует (кроме DB/DX).</li> <li>• Недостаточное значение для диапазона данных (кроме DB/DX).</li> </ul>	<p>Проверьте обладает ли партнер по связи требуемым диапазоном для данных и достаточен ли его размер. Проверьте параметры вызываемого SFB. Проверьте размер памяти для данных, определенный в SFB.</p>
W#16#0903	<p><b>Только для RK 512:</b> Ошибка доступа к DB/DX в станции партнера по связи (DB/DX не существуют или слишком малы). Для SIMATIC S5 как партнера по связи:</p> <ul style="list-style-type: none"> <li>• DB/DX не существуют.</li> <li>• DB/DX слишком малы.</li> <li>• Некорректный номер для DB/DX.</li> </ul> <p>Разрешенный диапазон памяти для передачи превышен при выполнении задания FETCH (выборка данных).</p>	<p>Проверьте обладает ли партнер по связи требуемым диапазоном для данных и достаточен ли его размер. Проверьте параметры вызываемого SFB. Проверьте размер памяти для данных, определенный в SFB.</p>
W#16#0904	<p><b>Только для RK 512:</b> Партнер по связи сообщает: "Job type not permitted" ("тип задания не разрешен").</p>	<p>Партнер по связи работает со сбоями, так как CPU никогда не выводит системных инструкций.</p>
W#16#0905	<p><b>Только для RK 512:</b> Ошибка в партнере по связи или ошибка в партнере SIMATIC S5:</p> <ul style="list-style-type: none"> <li>• Тип источника / приемника не разрешен.</li> <li>• Ошибка при работе с памятью в партнере по связи.</li> <li>• Ошибка в CP/CPU партнера по связи.</li> <li>• PLC партнера по связи находится в STOP-режиме.</li> </ul>	<p>Проверьте, способен ли партнер по связи передавать требуемые типы данных. Проверьте структуру оборудования партнера по связи. Переключите PLC партнера по связи в RAN-режим (режим выполнения).</p>

<b>"Получение ответного фрейма сообщения с ошибкой или фрейма сообщения об ошибках от партнера по связи"</b>		
<b>Код ошибки</b>	<b>Событие</b>	<b>Действия</b>
W#16#0908	<b>Только для RK 512:</b> Партнер распознает ошибку синхронизации: Последовательность фреймов сообщений прервана.	Эта ошибка происходит, когда Вы перезапускаете Ваш собственный PLC или PLC Вашего партнера по связи. Это - нормальное поведение системы при запуске. Никакие компенсационные мероприятия при этом не требуются. При выполнении задания такая ошибка могла бы быть следствием проявления предыдущих ошибок. Другими словами, Вы можете допускать подобное "сбойное" поведение партнера по связи.
W#16#0909	<b>Только для RK 512:</b> DB/DX в партнере по связи заблокированы меркером связи.	В программе партнера по связи: Сбросьте меркер связи после того, как Вы обработали последние переданные данные!  Ваша программа: повторно активируйте задание
W#16#090A	<b>Только для RK 512:</b> Партнером по связи распознаны ошибки в заголовке фрейма сообщения: • Неправильный байт команды в заголовке	Проверьте из-за чего происходит ошибка: в результате помех или из-за "сбойного" поведения партнера по связи. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#090C	<b>Только для RK 512:</b> Партнером по связи распознана ошибка, связанная с неправильной длиной фрейма сообщения (общей длиной).	Проверьте из-за чего происходит ошибка: в результате помех или из-за "сбойного" поведения партнера по связи. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#090D	<b>Только для RK 512:</b> До сих пор не произошло перезапуска партнера по связи.	Перезапустите PLC партнера по связи или установите переключатель в положение "RUN" ("выполнение").
W#16#090E	<b>Только для RK 512:</b> Принят неизвестный номер ошибки в ответном фрейме сообщения.	Проверьте из-за чего происходит ошибка: в результате помех или из-за "сбойного" поведения партнера по связи. Проверьте наличие сбоев в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.



<b>"CPU обнаружил ошибки в ответном фрейме партнера по связи"</b>		
<b>Код ошибки</b>	<b>Событие</b>	<b>Действия</b>
W#16#0A02	<b>Только для RK 512:</b> Ошибка в структуре принятого ответного фрейма сообщения. (Байт не 00 или FF)	Проверьте наличие сбоя в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0A03	<b>Только для RK 512:</b> Принятый ответный фрейм сообщения содержит слишком большое или слишком малое количество данных.	Проверьте наличие сбоя в поведении устройства партнера по связи, если требуется, подключая тестер интерфейса к каналу связи.
W#16#0A05	<b>Только для RK 512:</b> Нет ответного фрейма сообщения от партнера по связи на наблюдаемом периоде времени.	Возможно, партнером по связи является очень "медленное" устройство? Очень часто подобная ошибка идентифицируется в результате возникновения предыдущих ошибок. Например, при активной процедуре приема данных (класс события 8) ошибки могут отображаться после того, как передан фрейм сообщения о задании "FETCH" (выборка данных).  Причина: ответный фрейм сообщения не мог быть принят из-за помех, время цикла таймера (watchdog) истекает. Данная ошибка может также происходить, если партнер по связи был перезапущен до того, как он получил возможность ответить на последний принятый фрейм сообщения о задании "FETCH" (выборка данных).

<b>"Warnings" ("предупреждения")</b>		
<b>Код ошибки</b>	<b>Событие</b>	<b>Действия</b>
W#16#0B01	Входной буфер заполнен на 2/3 своего объема.	Вызывайте блок для приема более часто, чтобы избежать переполнения входного буфера.



## **27 SFC для H CPU**

## 27.1 Управление функционированием в H-системах с помощью SFC90 "H\_CTRL"

### Описание

С помощью SFC90 "H\_CTRL" Вы можете влиять на H-системы следующим образом:

- Вы можете предотвращать подключение резерва в главном CPU. Тогда оно блокируется до тех пор, пока Вы не отмените эту установку с помощью SFC90 "H\_CTRL".

Любой запрос из резервного CPU на соединение с главным сохраняется.

- Вы можете отключить обновление в главном CPU. Тогда оно блокируется до тех пор, пока Вы не отмените эту установку с помощью SFC90 "H\_CTRL".

Любой запрос из резервного CPU на обновление сохраняется.



### Предупреждение

Если Вы отключили возможность обновления, но оставили нетронутым соединение, то оборудование как прежде будет продолжать определять состояние соединения. При этом, если главный CPU подключен, он не будет обрабатывать никакие прерывания удаления/вставки модулей, прерывания отказа/восстановления станций или прерывания отказа/восстановления стоек.

- 
- Вы можете удалить компонент тестирования из системы циклического самотестирования, добавить его вновь или запустить на выполнение в любой момент.

### Примечание

Если Вы используете CPU 414-4H или 417-4H в системе с резервированием, то имейте в виду, что если Вы заблокируете компонент более, чем на 24 часа, то CPU перейдет в STOP-режим. Для систем с резервированием соответствующая система управления требует, чтобы окончательные тесты завершились в течение 24 часов.

---

Следующая таблица объясняет разрешенные комбинации входных параметров MODE и SUBMODE.

Задание	Вход MODE	Вход SUBMODE
Запретить соединение	3	0
Вновь разрешить соединение	4	0
Запретить обновление	1	0
Вновь разрешить обновление	2	0
Удалить компонент тестирования, определенный в SUBMODE из системы циклического самотестирования.	20	0,1,...5
Вновь вернуть компонент тестирования, определенный в SUBMODE в систему циклического самотестирования. Компонент тестирования может быть вновь возвращен в систему, только если он был ранее из нее удален.	21	0,1,...5
Запуск на выполнение задания тест-компонента, определенного в SUBMODE. Компонент тестирования не может быть при этом удаленным.	22	0,1,...5

В следующей таблице показано назначение отдельных тест-компонентов для циклического самотестирования со значениями входного параметра SUBMODE. (Соответствует только значениям 20, 21 и 22 на входе MODE)

Значение SUBMODE	Соответствующий компонент тестирования
0	SP7 – ASIC – Test
1	Тестирование раздела памяти кодов
2	Тестирование раздела памяти данных
3	Тестирование контрольной суммы кодов операционной системы
4	Тестирование контрольной суммы кодов блока
5	Сравнение числа, временных меток, меток и данных блоков при работе с резервированием

## Как работает SFC

SFC90 "H\_CTRL" является асинхронным SFC, иными словами, его выполнение может растягиваться на время нескольких вызовов SFC.

Задание запускают, вызывая SFC90 с REQ = 1.

Инициализация подпрограммы длительных испытаний заканчивается при первом вызове SFC (BUSY = 0), даже если тест производится в течение нескольких циклов (RET\_VAL = W#16#0001 для MODE = 22).

Если задание можно было выполнить сразу, то SFC возвращает в выходном параметре BUSY значение 0. Если BUSY имеет значение 1, то задание все еще активно. (См. раздел "Значения параметров REQ, RET\_VAL, BUSY для асинхронных SFC")

## Идентификация задания

Входные параметры MODE и SUBMODE определяют задание. Если они соответствуют заданию, которое еще не закончено, то вызов SFC является продолженным вызовом.

## Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
REQ	INPUT	BOOL	I, Q, M, D, L,	Запускаемый уровнем параметр управления REQ=1: запускает задание.
MODE	INPUT	BYTE	I, Q, M, D, L, константа	Задание.
SUBMODE	INPUT	BYTE	I, Q, M, D, L, константа	Подзадание.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	Если во время выполнения функции происходит ошибка, то возвращаемое значение содержит код ошибки. Обеспечьте проверку RET_VAL при каждом выполнении блока.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	BUSY=1: Задание еще не закончено.

## Информация об ошибках

Код ошибки (W#16#...)	Объяснение
0000	Задание выполнено без ошибки.
7000	REQ = 0 при первом вызове: задание не было активировано; BUSY имеет значение 0.
7001	REQ = 1 при первом вызове: задание было запущено; BUSY имеет значение 1.

Код ошибки (W#16#...)	Объяснение
7002	Продолженный вызов (REQ не имеет значения). Активированное задание все еще выполняется; BUSY имеет значение 1.
0001	<ul style="list-style-type: none"> <li>• MODE=1: обновление уже было заблокировано.</li> <li>• MODE=3: соединение уже было заблокировано.</li> <li>• MODE=22: компонент тестирования уже отработал и не может быть перезапущен.</li> </ul>
8082	<ul style="list-style-type: none"> <li>• MODE=1: обновление уже активно и больше не может блокироваться.</li> <li>• MODE=3: соединение уже активно и больше не может блокироваться.</li> <li>• MODE=20: компонент тестирования уже был удален из системы самотестирования.</li> <li>• MODE=21: компонент тестирования не удален из системы самотестирования.</li> <li>• MODE=22: компонент тестирования не может быть запущен, т.к. удален из системы самотестирования.</li> </ul>
8090	Входной параметр MODE имеет недопустимое значение.
8091	Входной параметр SUBMODE имеет недопустимое значение.

### Пример использования SFC 90

С помощью SFC90 "H\_CTRL" Вы можете добиться того, чтобы соединение и обновление не начинались в моменты времени, когда требуются максимальные ресурсы CPU.

Вы можете достичь этого включением следующих разделов в главном CPU перед периодом повышенной активности процесса:

- вызов SFC 90 с MODE = 3 и SUBMODE = 0 (запретить соединение)
- вызов SFC 90 с MODE = 1 и SUBMODE = 0 (запретить обновление).

В конце периода повышенной активности включите следующие разделы программы в главном CPU:

- вызов SFC 90 с MODE = 4 и SUBMODE = 0 (вновь разрешить соединение)
- вызов SFC 90 с MODE = 2 и SUBMODE = 0 (вновь разрешить обновление).





## **28 Встроенные функции (для CPU со встроенными входами/выходами)**

## 28.1 SFB29 (HS\_COUNT)

### Описание

С помощью SFB29 "HS\_COUNT" (счетчик) Вы можете воздействовать на встроенную функцию счетчика CPU со встроенными входами/выходами следующим образом:

- устанавливать и вводить начальное значение
- выбирать и устанавливать значения для сравнения
- разблокировать счетчики
- разблокировать цифровые выходы
- считывать текущие счетные значения и текущие значения для сравнения
- запрашивать соотношение между счетным значением и значением для сравнения.

### Дополнительная информация

Значение отдельных параметров SFB29 вместе с параметрами встроенной функции счетчика и аппаратных входов и выходов CPU подробно описано в руководстве *S7-300 Programmable Controller, Integrated Functions [Программируемый контроллер S7-300, встроенные функции]*.

### Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
PRES_COUNT	INPUT	DINT	I, Q, M, D, L, константа	Начальное значение счетчика
PRES_COMP_A	INPUT	DINT	I, Q, M, D, L, константа	Новое значение для сравнения COMP_A
PRES_COMP_B	INPUT	DINT	I, Q, M, D, L, константа	Новое значение для сравнения COMP_B
EN_COUNT	INPUT		I, Q, M, D, L	Разблокировка счетчика
EN_DO	INPUT	BOOL	I, Q, M, D, L, константа	Разблокировка цифровых выходов
SET_COUNT	INPUT	BOOL	I, Q, M, D, L, константа	Вход установки начального значения PRES_COUNT
SET_COMP_A	INPUT	BOOL	I, Q, M, D, L, константа	Вход установки значения для сравнения COMP_A
SET_COMP_B	INPUT	BOOL	I, Q, M, D, L, константа	Вход установки значения для сравнения COMP_B
COUNT	OUTPUT	DINT	I, Q, M, D, L	Фактическое значение счетчика
COMP_A	OUTPUT	DINT	I, Q, M, D, L	Текущее значение для сравнения COMP_A

Параметр	Описание	Тип данных	Область памяти	Характеристика
COMP_B	OUTPUT	DINT	I, Q, M, D, L	Текущее значение для сравнения COMP_B
STATUS_A	OUTPUT	BOOL	I, Q, M, D, L	Бит состояния STATUS_A 1: COUNT ≥ COMP_A 0: COUNT < COMP_A
STATUS_B	OUTPUT	BOOL	I, Q, M, D, L	Бит состояния STATUS_B 1: COUNT ≥ COMP_B 0: COUNT < COMP_B

## 28.2 SFB30 (FREQ\_MES)

### Описание

С помощью SFB30 "FREQ\_MES" (частотомер) Вы можете воздействовать на встроенную функцию частотомера CPU со встроенными входами/выходами следующим образом:

- выбирать и устанавливать значения для сравнения
- выводить измеренную частоту
- считывать текущие значения для сравнения
- запрашивать отношение измеренной частоты к значению для сравнения.

### Дополнительная информация

Значение отдельных параметров SFB30 вместе с параметрами встроенной функции частотомера и аппаратных входов и выходов CPU подробно описано в руководстве *S7-300 Programmable Controller, Integrated Functions [Программируемый контроллер S7-300, встроенные функции]*.

### Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
PRES_U_LIMIT	INPUT	DINT	I, Q, M, D, L, константа	Новое (верхнее) значение для сравнения U_LIMIT
PRES_L_LIMIT	INPUT	DINT	I, Q, M, D, L, константа	Новое (нижнее) значение для сравнения L_LIMIT
SET_U_LIMIT	INPUT	BOOL	I, Q, M, D, L, константа	Вход установки нового значения для сравнения U_LIMIT
SET_L_LIMIT	INPUT	BOOL	I, Q, M, D, L, константа	Вход установки нового значения для сравнения L_LIMIT
FREQ	OUTPUT	DINT	I, Q, M, D, L	Измеренная частота в МГц
U_LIMIT	OUTPUT	DINT	I, Q, M, D, L	Текущее значение для сравнения (верхний предел)
L_LIMIT	OUTPUT	DINT	I, Q, M, D, L	Текущее значение для сравнения (нижний предел)
STATUS_U	OUTPUT	BOOL	I, Q, M, D, L	Бит состояния "1": FREQ > U_LIMIT "0": FREQ ≤ U_LIMIT
STATUS_L	OUTPUT	BOOL	I, Q, M, D, L	Бит состояния "1": FREQ < L_LIMIT "0": FREQ ≥ U_LIMIT

## 28.3 SFB38 (HSC\_A\_B)

### Описание

С помощью SFB38 (HSC\_A\_B) Вы можете воздействовать на встроенную функцию счетчика A/B в CPU со встроенными входами/выходами, следующим образом:

- задавать и принимать начальное значение
- задавать и устанавливать значения для сравнения
- разблокировать счетчики
- разблокировать цифровые выходы
- считывать текущие счетные значения и текущие значения для сравнения.
- запрашивать соотношение между счетным значением и значением для сравнения

SFB38 (HSC\_A\_B) читает или записывает данные из программы пользователя в экземплярном DB встроенной функции. Счетчик A/B состоит из двух счетчиков A и B, которые могут вести счет одновременно и независимы друг от друга (возможен прямой и обратный счет). Счетчики функционируют одинаково; могут регистрироваться счетные импульсы частотой до 10 кГц.

### Дополнительная информация

Точное значение параметров SFB38 вместе с параметрами встроенной функции счетчика A/B и аппаратных входов и выходов CPU подробно описано в руководстве *S7-300 Programmable Controller, Integrated Functions CPU 312 IFM/314 IFM* [Программируемый контроллер S7-300, встроенные функции CPU 312 IFM/314 IFM].

### Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
PRES_COMP	INPUT	DINT	I, Q, M, D, L, константа	Новое значение для сравнения COMP
EN_COUNT	INPUT	BOOL	I, Q, M, D, L константа	Разблокировка счетчика
EN	INPUT	BOOL	I, Q, M, D, L, константа	Разблокировка цифрового выхода
SET_COMP	INPUT	BOOL	I, Q, M, D, L, константа	Вход установки значения для сравнения COMP
COUNT	OUTPUT	DINT	I, Q, M, D, L	Фактическое значение счетчика
COMP	OUTPUT	DINT	I, Q, M, D, L	Текущее значение для сравнения COMP
ENO	OUTPUT	BOOL	I, Q, M, D, L	Обработка ошибок: 1 : нет ошибки при выполнении 0 : ошибка при выполнении

## 28.4 SFB39 (POS)

### Описание

С помощью SFB39 (POS) Вы можете воздействовать на встроенную функцию позиционирования CPU со встроенными входами/выходами. SFB39 (POS) предоставляет в распоряжение следующие функции:

- Синхронизация
- Выполнение замедленного режима
- Позиционирование

SFB39 (POS) для встроенной функции позиционирования читает или записывает данные из программы пользователя в экземпляр DB встроенной функции. Встроенная функция позиционирования собирает сигналы от асимметричных инкрементальных шифраторов с напряжением 24 В с частотой до 10 кГц. Она управляет режимом «быстро/медленно» или преобразователем частоты через заданные встроенные выходы CPU 314 IFM (управляемое позиционирование).

### Дополнительная информация

Точное значение параметров SFB39 вместе с параметрами встроенной функции счетчика A/B и аппаратных входов и выходов CPU подробно описано в руководстве *S7-300 Programmable Controller, Integrated Functions CPU 312 IFM/314 IFM [Программируемый контроллер S7-300, встроенные функции CPU 312 IFM/314 IFM]*.

### Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
EN	INPUT	BOOL	I, Q, M, D, L, константа	Разблокировка цифровых выходов
DEST_VAL	INPUT	DINT	I, Q, M, D, L, константа	Целевая позиция для интегрированной функции позиционирования
REF_VAL	INPUT	DINT	I, Q, M, D, L, константа	Опорная точка синхронизации
SWITCH_OFF_DIFF	INPUT	WORD	I, Q, M, D, L, константа	Упреждение выключения (разность между точкой выключения и целевой позицией) в шагах перемещения
PRES_COMP	INPUT	DINT	I, Q, M, D, L, константа	Новое значение для сравнения COMP
BREAK	INPUT	BYTE	I, Q, M, D, L, константа	Максимальное аналоговое значение, которым регулируется поперечное движение

POS_MODE1, POS_MODE2	INPUT	BOOL	I, Q, M, D, L, константа	Запуск и выполнение замедленного режима
POS_STRT	INPUT	BOOL	I, Q, M, D, L, константа	Запуск операции позиционирования при нарастающем фронте
SET_POS	INPUT	BOOL	I, Q, M, D, L, константа	При нарастающем фронте значение входного параметра REF_VAL принимается как новое фактическое значение
ENO	OUTPUT	BOOL	I, Q, M, D, L	Обработка ошибок: 1 : нет ошибки при выполнении 0 : ошибка при выполнении
ACTUAL_POS	OUTPUT	DINT	I, Q, M, D, L	Текущее фактическое значение
POS_READY (сообщение о состоянии)	OUTPUT	BOOL	I, Q, M, D, L	Позиционирование/ замедленное перемещение завершено, если POS_READY=1
REF_VALID (сообщение о состоянии)	OUTPUT	BOOL	I, Q, M, D, L	Достигнут или нет переключатель опорной точки
POS_VALID (сообщение о состоянии)	OUTPUT	BOOL	I, Q, M, D, L	Фактическое положение оси синхронизировано с фактическим положением встроенной функции





## 29 Гибкая технология

## 29.1 SFC63 (AB\_CALL)

### Описание

SFC63 (AB\_CALL) вызывает блок скомпонованного кода.

Блоки скомпонованного кода – это логические блоки, которые были написаны на языке программирования С или на Ассемблере, а затем скомпилированы.

### Применение

Вы можете использовать блоки скомпонованного кода только для CPU 614.

### Дополнительная информация

Значение отдельных параметров SFC63 подробно объясняется в документации для CPU 614. Имеется отдельное руководство по программированию блоков скомпонованного кода.

### Параметры

Параметр	Описание	Тип данных	Область памяти	Характеристика
AB_NUMBER	INPUT	WORD	I, Q, M, D, L, константа	Биты для вызываемых блоков скомпонованного кода
CALL_REASON	INPUT	WORD	I, Q, M, D, L, константа	Организационный блок, в котором вызывался SFC, или проверка указателя DB (параметр DB_NUMBER), или активация отладчика
DB_NUMBER	INPUT	WORD	I, Q, M, D, L, константа	Номер указателя DB
RET_VAL	OUTPUT	WORD	I, Q, M, D, L	Возвращаемое значение SFC

## **30 Диагностические данные**

## 30.1 Обзор структуры диагностических данных

### Записи данных 0 и 1 области системных данных

Диагностические данные модуля находятся в записях данных 0 и 1 области системных данных (см. раздел 7.1).

- Запись данных 0 содержит 4 байта диагностических данных, которые описывают текущее состояние сигнального модуля.
- Запись данных 1 содержит
  - 4 байта диагностических данных, которые находятся также в записи данных 0, и
  - диагностические данные, специфические для модуля.

### Структура и содержание диагностических данных

Этот раздел описывает структуру и содержание отдельных байтов диагностических данных.

Если появляется ошибка, то соответствующий бит устанавливается в "1".

## 30.2 Диагностические данные

Структура и содержание диагностических данных:

Байт	Бит	Значение	Примечание
0	0	Неисправность модуля	
	1	Внутренняя ошибка	
	2	Внешняя ошибка	
	3	Ошибка канала	
	4	Отсутствует внешнее вспомогательное напряжение	
	5	Отсутствует передний штепсельный разъем	
	6	Отсутствует параметризация	
	7	Неправильные параметры в модуле	
1	0 ... 3	Класс модуля	0101 Аналоговый модуль
			0000 CPU
			1000 Функциональный модуль
			1100 СР
			1111 Цифровой модуль
			0011 Стандартное slave-устройство DP
			1011 I-slave
			0100 IM
	4	Имеется информация канала	
	5	Имеется информация пользователя	
	6	Диагностическое прерывание из-за замены	
	7	Резерв	
2	0	Модуль памяти неисправен или отсутствует	
	1	Нарушение связи	
	2	Рабочий режим	0 RUN 1 STOP
	3	Сработал контроль времени цикла	
	4	Отказ внутреннего модуля питания	
	5	Разрядилась батарея	
	6	Отказ всей батарейной буферизации	
	7	Резерв	
3	0	Отказ стойки расширения	
	1	Отказ процессора	
	2	Ошибка EPROM	
	3	Ошибка RAM	
	4	Ошибка АЦП/ЦАП	
	5	Сгорел предохранитель	
	6	Потеря аппаратного прерывания	
	7	Резерв	

Байт	Бит	Значение	Примечание
4	0 ... 6	Тип канала	В#16#70 Цифровой ввод В#16#72 Цифровой вывод В#16#71 Аналоговый ввод В#16#73 Аналоговый вывод В#16#74 FM-POS В#16#75 FM-REG В#16#76 FM-ZAENL В#16#77 FM-TECHNO В#16#78 FM-NCU В#16#79 ... ... В#16#7D резерв В#16#7E US300 В#16#7F резерв
	7	Есть дополнительный канал?	0 нет 1 да
5	0 ... 7	Количество битов диагностики, выводимых модулем на один канал.	Количество битов диагностики на один канал округляется до границ байта
6	0 ... 7	Количество каналов одного типа на модуль	Если в модуле существуют разные типы каналов, то структура для каждого типа канала повторяется, начиная с байта 4 в записи данных 1.
7	0	Ошибка канала - канал 0/ группа каналов 0	Первый байт вектора ошибок канала (длина вектора ошибок канала определяется числом каналов и округляется до границ байта).
	1	Ошибка канала - канал 1/ группа каналов 1	
	2	Ошибка канала - канал 2/ группа каналов 2	
	3	Ошибка канала - канал 3/ группа каналов 3	
	4	Ошибка канала - канал 4/ группа каналов 4	
	5	Ошибка канала - канал 5/ группа каналов 5	
	6	Ошибка канала - канал 6/ группа каналов 6	
	7	Ошибка канала - канал 7/ группа каналов 7	
...	-	Специфические для канала ошибки (см. раздел "Структура диагностических данных, относящихся к каналу")	

### 30.3 Структура диагностических данных, относящихся к каналу

#### Ошибки, относящиеся к каналу

Начиная с байта, который находится непосредственно за вектором ошибок каналов, для каждого канала модуля отображаются относящиеся к этому каналу ошибки. Следующая таблица показывает структуру относящихся к каналу диагностических данных для разных типов каналов. Биты имеют следующие значения:

- 1 = ошибка
- 0 = нет ошибки

#### Канал ввода аналоговых сигналов

Диагностический байт канала аналогового ввода:

Бит	Значение	Примечание
0	Ошибка конфигурирования/ параметризации	Может быть передана функцией SFC52 и EVENTN = W#16#8x50
1	Синфазная ошибка	Может быть передана функцией SFC52 и EVENTN = W#16#8x51
2	Короткое замыкание на Р	Может быть передана функцией SFC52 и EVENTN = W#16#8x52
3	Короткое замыкание на М	Может быть передана функцией SFC52 и EVENTN = W#16#8x53
4	Обрыв провода	Может быть передана функцией SFC52 и EVENTN = W#16#8x54
5	Ошибка опорного канала	Может быть передана функцией SFC52 и EVENTN = W#16#8x55
6	Ток ниже границы диапазона измерения	Может быть передана функцией SFC52 и EVENTN = W#16#8x56
7	Ток выше границы диапазона измерения	Может быть передана функцией SFC52 и EVENTN = W#16#8x57

**Канал аналогового вывода**

Диагностический байт для канала аналогового вывода:

Бит	Значение	Примечание
0	Ошибка конфигурирования/ параметризации	Может быть передана функцией SFC52 и EVENTN = W#16#8x60
1	Синфазная ошибка	Может быть передана функцией SFC52 и EVENTN = W#16#8x61
2	Короткое замыкание на Р	Может быть передана функцией SFC52 и EVENTN = W#16#8x62
3	Короткое замыкание на М	Может быть передана функцией SFC52 и EVENTN = W#16#8x63
4	Обрыв провода	Может быть передана функцией SFC52 и EVENTN = W#16#8x64
5	0	резерв
6	Отсутствует напряжение нагрузки	Может быть передана функцией SFC52 и EVENTN = W#16#8x66
7	0	резерв



**Канал дискретного ввода**

Диагностический байт для канала дискретного ввода:

Бит	Значение	Примечание
0	Ошибка конфигурирования/ параметризации	Может быть передана функцией SFC52 и EVENTN = W#16#8x70
1	Ошибка заземления на корпус	Может быть передана функцией SFC52 и EVENTN = W#16#8x71
2	Короткое замыкание на Р (датчик)	Может быть передана функцией SFC52 и EVENTN = W#16#8x72
3	Короткое замыкание на М	Может быть передана функцией SFC52 и EVENTN = W#16#8x73
4	Обрыв провода	Может быть передана функцией SFC52 и EVENTN = W#16#8x74
5	Отсутствует питание датчика	Может быть передана функцией SFC52 и EVENTN = W#16#8x75
6	0	резерв
7	0	резерв

**Канал дискретного вывода**

Диагностический байт для канала дискретного вывода:

Бит	Значение	Примечание
0	Ошибка конфигурирования/ параметризации	Может быть передана функцией SFC52 и EVENTN = W#16#8x80
1	Ошибка заземления на корпус	Может быть передана функцией SFC52 и EVENTN = W#16#8x81
2	Короткое замыкание на Р	Может быть передана функцией SFC52 и EVENTN = W#16#8x82
3	Короткое замыкание на М	Может быть передана функцией SFC52 и EVENTN = W#16#8x83
4	Обрыв провода	Может быть передана функцией SFC52 и EVENTN = W#16#8x84
5	Сгорел предохранитель	Может быть передана функцией SFC52 и EVENTN = W#16#8x86
6	Отсутствует напряжение нагрузки	Может быть передана функцией SFC52 и EVENTN = W#16#8x86
7	Перегрев	Может быть передана функцией SFC52 и EVENTN = W#16#8x87



## **31 Списки состояний системы (SSL)**

## 31.1 Обзор списков состояний системы (SSL)

### Обзор этой главы

Эта глава описывает все подписки списка состояний системы, которые относятся:

- к CPU
- к модулям, подписки которых не являются специфическими для модулей (например, SSL-ID W#16#00B1, W#16#00B2, W#16#00B3).

Специфические для модулей подписки, в частности, для CP и FM включены в описания конкретных модулей.

### Определение: Список состояний системы

Список состояний системы (system status list, SSL) описывает текущее состояние программируемого логического контроллера. Содержание SSL может быть только прочитано с помощью информационных функций, но не изменено. Подписки являются виртуальными списками, то есть они создаются операционной системой центральных модулей только конкретному по запросу.

С помощью SFC 51 "RDSYSST" Вы можете прочитать только один список состояний системы.

### Содержание

Список состояний системы содержит информацию о:

- системных данных
- данных состояния диагностики в CPU
- диагностических данных модулей
- диагностическом буфере

### Системные данные

Системные данные представляют собой фиксированные или назначенные характеристики CPU. Они предоставляют информацию:

- о конфигурации CPU
- о состоянии классов приоритета
- о связи

### Данные состояния диагностики

Данные состояния диагностики описывают текущее состояние компонентов, контролируемых системными диагностическими функциями.

### Диагностические данные модулей

Поставленные в соответствие CPU модули с диагностическими способностями обладают диагностическими данными, которые хранятся непосредственно в модуле.

### Диагностический буфер

Диагностический буфер содержит диагностические записи в порядке их появления.

## 31.2 Структура подписка SSL

### Основы

Вы можете прочитать подписание или фрагменты подписка с помощью SFC51 "RDSYSST". Что Вы хотите прочитать, Вы указываете с помощью параметров SSL-ID и INDEX.

### Структура

Подписание состоит из:

- заголовка и
- записей данных.

### Заголовок

Заголовок подписка состоит из:

- SSL-ID (идентификатора SSL)
- индекса
- длины записи данных этого подписка в байтах
- количества записей данных, содержащихся в этом подписке.

### Индекс

С конкретными подписками или фрагментами подписков должен быть задан идентификатор типа объекта или номер объекта. Для этого используется индекс. Если он не нужен для информации, то его содержание не имеет значения.

### Записи данных

Запись данных подписка имеет определенную длину. Она зависит от информации, содержащейся в подписке. Как используются слова данных в записи данных, также зависит от конкретного подписка.

## 31.3 SSL-ID

### SSL-ID

Каждый подсписок состояний системы имеет номер. Вы можете вывести полный подсписок или фрагмент из него. Возможные фрагменты подсписков predeterminedены и идентифицируются с помощью номера. SSL-ID (идентификатор SSL) состоит из номера подсписка, номера фрагмента подсписка и класса модуля.

### Структура

Длина SSL-ID равна одному слову. Биты SSL-ID имеют следующие значения:



Структура SSL-ID

### Класс модуля

Примеры классов модулей:

Класс модуля	Код (двоичный)
CPU	0000
IM	0100
FM	1000
CP	1100

### Номер фрагмента подписка

Номера фрагментов подписка и их значение зависят от конкретного списка состояний системы, которому они принадлежат. С помощью номера фрагмента подписка Вы задаете, какое подмножество подписка Вы хотите прочитать.

### Номер подписка

С помощью номера подписка Вы задаете, какой подсписок SSL Вы хотите прочитать.

## 31.4 Возможные подписки состояний системы

### Подмножество

Любой модуль содержит только подмножество всех возможных подписков. Какие подписки имеются в распоряжении, зависит от конкретного модуля.

### Возможные подписки SSL

В следующей таблице представлены все возможные подписки с соответствующими номерами в SSL-ID.

Подсписок	SSL-ID
Идентификация модуля	W#16#xy11
Характеристики CPU	W#16#xy12
Области памяти пользователя	W#16#xy13
Системные области	W#16#xy14
Типы блоков	W#16#xy15
Состояние светодиодов модуля	W#16#xy19
Состояние прерываний	W#16#xy22
Данные о состоянии связи	W#16#xy32
Групповая информация H CPU	W#16#xy71
Состояние светодиодов модуля	W#16#xy74
Включенные ведомые устройства DP в H-системах	W#16#xy75
Информация о состоянии модуля	W#16#xy91
Информация о состоянии стойки/ станции	W#16#xy92
Диагностический буфер CPU	W#16#xyA0
Диагностическая информация модуля (запись данных 0)	W#16#00B1
Диагностическая информация модуля (запись данных 1), географический адрес	W#16#00B2
Диагностическая информация модуля (запись данных 1), логический адрес	W#16#00B3
Диагностические данные ведомые DP-устройства	W#16#00B4

## 31.5 SSL-ID W#16#xy11 – Идентификация модуля

### Цель

Если Вы читаете список состояний системы с SSL-ID W#16#xy11, то Вы получаете идентификацию этого модуля.

### Заголовок

Заголовок списка состояний системы с SSL-ID W#16#xy11 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подсписка W#16#0111: отдельная идентификационная запись данных
INDEX	Номер конкретной записи данных W#16#0001: идентификация модуля W#16#0006: идентификация базового аппаратного обеспечения W#16#0007: идентификация базового микропрограммного обеспечения
LENGTHDR	W#16#001C: одна запись данных имеет длину 14 слов (28 байтов)
N_DR	Количество записей данных

### Запись данных

Запись данных списка состояний системы с SSL-ID W#16#xy11 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	Индекс идентификационной записи данных
MIFB	20 байтов	Индекс W#16#0007: резерв Индекс W#16#0001 и W#16#0006: порядковый номер модуля; строка содержит 19 символов и пробелов (20H); например, для CPU 314: "6ES7 314-0AE01-0AB0"
BGTyp	1 слово	Резерв
Ausbg1	1 слово	Индекс W#16#0001: версия модуля Индекс W#16#0006 и W#16#0007: "V" и первая цифра идентификатора версии (ID)
Ausbg2	1 слово	Индекс W#16#0001: резерв Индекс W#16#0006 и W#16#0007: оставшиеся цифры идентификатора версии



## 31.6 SSL-ID W#16#ху12 – Характеристики CPU

### Цель

Модули CPU обладают различными характеристиками в зависимости от используемого аппаратного обеспечения. Каждой характеристике поставлен в соответствие идентификатор. Если Вы читаете подписание с SSL-ID W#16#ху12, то Вы получаете характеристики модуля.

### Заголовок

Заголовок подписка SSL-ID W#16#ху12 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подписка: W#16#0012: все характеристики W#16#0112: характеристики группы Группу Вы задаете в параметре INDEX. W#16#0F12: информация заголовка подписка
INDEX	Группа W#16#0000: блок обработки MC7 W#16#0100: система времени W#16#0200: реакция системы W#16#0300: описание языка MC7 W#16#0400: доступные SFC
LENGTHDR	W#16#0002: одна запись данных имеет длину 1 слово (2 байта)
N_DR	Количество записей данных

### Запись данных

Запись данных подписка с SSL-ID W#16#ху12 имеет длину одно слово. Идентификатор введен для каждой характеристики. Идентификатор характеристики имеет длину одно слово.

### Идентификатор характеристики

Следующая таблица перечисляет все идентификаторы характеристик.

Идентификатор	Значение
W#16#0000 - 00FF	Блок обработки MC7 (группа с индексом 0000)
W#16#0001	Обработка MC7, генерирующая код
W#16#0002	Интерпретатор MC7
W#16#0100 - 01FF	Система времени (группа с индексом 0100)
W#16#0101	Разрешающая способность 1 мс
W#16#0102	Разрешающая способность 10 мс
W#16#0103	Нет часов реального времени
W#16#0104	BBCD-формат времени суток
W#16#0200 - 02FF	Реакция системы (группа с индексом 0200)

Идентификатор	Значение
W#16#0201	Способность к многопроцессорной обработке
W#16#0202	
W#16#0203	
W#16#0204	
W#16#0205	
W#16#0300 - 03FF	Описание языка MC7 CPU (группа с индексом 0300)
W#16#0301	резерв
W#16#0302	Все 32-битовые команды вычислений с фиксированной точкой
W#16#0303	Все команды вычислений с плавающей точкой
W#16#0304	sin,asin,cos,acos,tan,atan,sqr,sqrt,ln,exp
W#16#0305	Аккумулятор 3/аккумулятор 4 с соответствующими командами (ENT,PUSH,POP,LEAVE)
W#16#0306	Команды главного управляющего реле (Master Control Relay)
W#16#0307	Имеется адресный регистр 1 с соответствующими командами
W#16#0308	Имеется адресный регистр 2 с соответствующими командами
W#16#0309	Операции для адресации, выходящей за пределы области
W#16#030A	Операции для адресации внутри области
W#16#030B	Все команды косвенной адресации через память для памяти с побитовым доступом (меркеры) (M)
W#16#030C	Все команды косвенной адресации через память для блоков данных (DB)
W#16#030D	Все команды косвенной адресации через память для блоков данных (DI)
W#16#030E	Все команды косвенной адресации через память для локальных данных (L)
W#16#030F	Все команды передачи параметров в FC
W#16#0310	Команды, связанные с фронтом бита памяти (меркера), для входа образа процесса (I)
W#16#0311	Команды, связанные с фронтом бита памяти, для выхода образа процесса (Q)
W#16#0312	Команды, связанные с фронтом бита памяти, для бита памяти (M)
W#16#0313	Команды, связанные с фронтом бита памяти, для блоков данных (DB)
W#16#0314	Команды, связанные с фронтом бита памяти, для блоков данных (DI)
W#16#0315	Команды, связанные с фронтом бита памяти, для локальных данных (L)
W#16#0316	Динамическая оценка бита FC
W#16#0317	Динамическая область локальных данных с соответствующими командами
W#16#0318	Резерв
W#16#0319	Резерв
W#16#0401	Функция SFC 87 "C_DIAG" доступна
W#16#0402	Функция SFC 88 "C_CNTRL" доступна

## 31.7 SSL-ID W#16#ху13 – Области памяти

### Цель

Считывая подпсик с SSL-ID W#16#ху13, Вы получаете информацию об областях памяти модуля.

### Заголовок

Заголовок подпсика SSL-ID W#16#ху13 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подпсика W#16#0113: запись данных для одной области памяти Вы задаете область памяти через параметр INDEX.
INDEX	Задаёт область памяти (только с SSL-ID W#16#0113) W#16#0001: рабочая память
LENGTHDR	W#16#0024: одна запись данных имеет длину 18 слов (36 байтов)
N_DR	Количество записей данных

### Запись данных

Запись данных подпсика с SSL-ID W#16#ху13 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	Индекс области памяти W#16#0001: рабочая память
Code	1 слово	Тип памяти: W#16#0001: энергозависимая память (RAM) W#16#0002: энергонезависимая память (FEPROM) W#16#0003: смешанная память (RAM + FEPROM)
Size	2 слова	Общий размер выбранной памяти (сумма области 1 и области 2)
Mode	1 слово	Логический тип памяти Бит 0: энергозависимая область памяти Бит 1: энергонезависимая область памяти Бит 2: смешанная область памяти Для рабочей памяти: Бит 3: код и данные отдельно Бит 4: код и данные отдельно
Granu	1 слово	Всегда имеет значение 0
Ber1	2 слова	Размер энергозависимой области памяти в байтах.
Belegt1	2 слова	Размер используемой энергозависимой области памяти в байтах
Block1	2 слова	Наибольший свободный блок в энергозависимой области памяти Если 0: информация отсутствует или не может быть найдена.
Ber2	2 слова	Размер энергонезависимой области памяти в байтах
Belegt2	2 слова	Размер занятой энергонезависимой области памяти
Block2	2 слова	Наибольший свободный блок в энергонезависимой области памяти Если 0: информация отсутствует или не может быть найдена.

## 31.8 SSL-ID W#16#ху14 – Системные области

### Цель

Считывая подсписк с SSL-ID W#16#ху14, Вы получаете информацию о системных областях модуля.

### Заголовок

Заголовок подсписка с SSL-ID W#16#ху14 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подсписка W#16#0014: все системные области модуля W#16#0F14: только для информации заголовка подсписка
INDEX	Не имеет значения
LENGTHDR	W#16#0008: на запись данных имеет длину 4 слова (8 байтов)
N_DR	Количество записей данных. Вы должны назначить по крайней мере число из 9 записей данных. Если выбрать область назначения, которая окажется слишком мала, то функция SFC51 не сможет обеспечить запись данных.

### Запись данных

Запись данных подсписка с SSL-ID W#16#ху14 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	<p>Индекс системной области</p> <p>W#16#0001: PII (количество в байтах)</p> <p>W#16#0002: PIQ (количество в байтах)</p> <p>W#16#0003: память (меркеры) (количество в битах)</p> <p><b>Примечание:</b> Данный индекс обеспечивается только в том CPU, в котором число меркеров (флагов) может быть показано одним словом. Если Ваш CPU не обеспечивает это значение, Вы должны проверить индекс W#16#0008.</p> <p>W#16#0004: таймеры (количество)</p> <p>W#16#0005: счетчики (количество)</p> <p>W#16#0006: количество байтов в логическом адресном пространстве</p> <p>W#16#0007: локальные данные (вся область локальных данных CPU в байтах)</p> <p><b>Примечание:</b> Данный индекс обеспечивается только в том CPU, в котором число меркеров (флагов) может быть показано одним словом. Если Ваш CPU не обеспечивает это значение, Вы должны проверить индекс W#16#0009.</p> <p>W#16#0008: память (меркеры) (количество в байтах)</p> <p>W#16#0009: локальные данные (вся область локальных данных CPU в килобайтах)</p>

<b>Имя</b>	<b>Длина</b>	<b>Значение</b>
Code	1 слово	Тип памяти W#16#0001: энергозависимая память (RAM) W#16#0002: энергонезависимая память (EEPROM) W#16#0003: смешанная память (RAM и EEPROM)
Quantity	1 слово	Количество элементов системной области
Reman	1 слово	Количество сохраняемых (перманентных) элементов

## 31.9 SSL-ID W#16#xy15 – Типы блоков

### Цель

Считывая подсписк с SSL-ID W#16#xy15, Вы получаете типы блоков, которые имеются в модуле.

### Заголовок

Заголовок подсписка SSL-ID W#16#xy15 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подсписка W#16#0015: Записи данных всех типов блоков модуля
INDEX	Не имеет значения
LENGTHDR	W#16#0006: одна запись данных имеет длину 5 слов (10 байтов)
N_DR	Количество записей данных

### Запись данных

Запись данных подсписка с SSL-ID W#16#xy15 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	Номер типа блока W#16#0800: OB W#16#0A00: DB W#16#0B00: SDB W#16#0C00: FC W#16#0E00: FB
MaxAnz	1 слово	Максимальное количество блоков этого типа OB: максимально возможное количество OB в CPU DB: максимально возможное количество DB, включая DB0 SDB: максимально возможное количество SDB, включая SDB2 FC и FB: максимально возможное количество загружаемых блоков
MaxLng	1 слово	Максимальный общий размер загружаемых объектов в Кбайтах
Maxabl	2 слова	Максимальная длина раздела рабочей памяти блока в байтах

## 31.10 SSL-ID W#16#ху19 – Состояние светодиодов модуля

### Цель

Считывая подсписк с SSL-ID W#16#ху19, Вы получаете состояние светодиодов модуля.

---

### Примечание

Если Вы хотите считать подсписк W#16#16#ху19 для H CPU, помните, что это возможно только в не резервируемых режимах работы H.

---

### Заголовок

Заголовок подсписка W#16#ху19 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID подсписка списка состояний системы W#16#0019 состояние всех светодиодов W#16#0119 состояние одного светодиода
LENGTHDR	W#16#0004: одна запись данных имеет длину 2 слова (4 байта)
N_DR	Количество записей данных

**Запись данных**

Запись данных подсписка с SSL-ID W#16#ху19 имеет следующую структуру:

Имя	Длина	Значение
INDEX	1 слово	LED ID (имеет значение только для SSL-ID W#16#0119) W#16#0001: SF (групповая ошибка) W#16#0002: INTF (внутренняя ошибка) W#16#0003: EXTf (внешняя ошибка) W#16#0004: RUN W#16#0005: STOP W#16#0006: FRCE (принудительное задание) W#16#0007: CRST (рестарт) W#16#0008: BAF (неисправность батареи/перегрузка, короткое замыкание напряжения батареи на шину) W#16#0009: USR (определено пользователем) W#16#000A: USR1 (определено пользователем) W#16#000B: BUS1F (ошибка шины, интерфейс 1) W#16#000C: BUS2F (ошибка шины, интерфейс 2) W#16#000D: REDF (ошибка резервирования) W#16#000E: MSTR (мастер) W#16#000F: RACK0 (стойка номер 0) W#16#0010: RACK1 (стойка номер 1) W#16#0011: RACK2 (стойка номер 2) W#16#0012: IFM1F (ошибка интерфейса интерфейсного модуля 1) W#16#0013: IFM2F (ошибка интерфейса интерфейсного модуля 2)
led_on	1 байт	Состояние светодиода: 0 : выключено 1 : включено
led_blink	1 байт	Состояние мигания светодиода: 0: не мигает 1: мигает с нормальной частотой (2 Гц) 2: мигает с пониженной частотой (0,5 Гц)



## 31.11 SSL-ID W#16#xy1C – Идентификация компонентов

### Цель

Считывая подписание с SSL-ID W#16#xy1C, Вы можете идентифицировать CPU или PLC.

### Заголовок

Заголовок подписка W#16#xy1C построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подписка W#16#001C: Идентификация всех компонентов W#16#011C: Идентификация одного компонента W#16#0F1C: Только информация заголовка подписка
INDEX	Идентификация компонента для подписка SSL при SSL-ID W#16#011C W#16#0001: Имя PLC W#16#0002: Имя CPU W#16#0003: Идентификатор изготовителя CPU W#16#0004: Копирайт Siemens W#16#0006: Резерв для операционной системы
LENGTHDR	W#16#0022: одна запись данных имеет длину 17 слов (34 байта)
N_DR	Количество записей данных

**Запись данных**

Запись данных подсписка с SSL-ID W#16#ху1С имеет следующую структуру:

- INDEX = W#16#0001

Имя	Длина	Значение
INDEX	1 слово	Идентификация компонента: W#16#0001
name	12 слов	Имя PLC (максимум 24 символа; при использовании более коротких имен вставляются пробелы с В#16#00)
res	4 слова	Резерв

- INDEX = W#16#0002

Имя	Длина	Значение
INDEX	1 слово	Идентификация компонента: W#16#0002
name	12 слов	Имя CPU (максимум 24 символа; при использовании более коротких имен вставляются пробелы с В#16#00)
res	4 слова	Резерв

- INDEX = W#16#0003

Имя	Длина	Значение
INDEX	1 слово	Идентификация компонента: W#16#0003
tag	16 слов	Идентификация изготовителя CPU (максимум 32 символа; при использовании более коротких имен вставляются пробелы с В#16#00)

- INDEX = W#16#0004

Имя	Длина	Значение
INDEX	1 слово	Идентификация компонента: W#16#0004
copyright	13 слов	Символьная строка-константа "Original Siemens Equipment"
res	3 слова	Резерв

- INDEX = W#16#0006

Соответствующая запись данных зарезервирована за ОС.

## 31.12 SSL-ID W#16#xy22 – Состояние прерывания

### Цель

Считывая подпоскок с SSL-ID W#16#xy22, Вы получаете информацию о текущем состоянии обработки прерывания и о прерываниях, генерируемых модулем.

### Заголовок

Заголовок подпоскока с SSL-ID W#16#xy22 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID подпоскока списка состояний системы W#16#0222 запись данных для заданного прерывания. Вы задаете прерывание (номер ОБ) с помощью параметра INDEX.
INDEX	Класс прерывания или номер ОБ (при SSL-ID W#16#0222) W#16#0000: свободный цикл W#16#0A0A: прерывание по времени W#16#1414: прерывание с задержкой W#16#1E23: циклическое прерывание W#16#2828: аппаратное прерывание W#16#5050: прерывание по асинхронной ошибке W#16#005A: фоновое W#16#0064: запуск W#16#7878: прерывание по синхронной ошибке
LENGTHDR	W#16#001C: одна запись данных имеет длину 14 слов (28 байтов)
N_DR	Количество записей данных

### Запись данных

Запись данных подпоскока с SSL-ID W#16#xy22 имеет следующую структуру:

Имя	Длина	Значение
info	10 слов	<p>Стартовая информация для соответствующего ОВ со следующими исключениями:</p> <ul style="list-style-type: none"> <li>Для ОВ1 имеются в распоряжении текущее минимальное (в байтах 8 и 9) и максимальное (в байтах 10 и 11) время цикла (база времени: мс, отсчет байтов начинается с 0).</li> <li>Для ОВ80 сконфигурированные минимальное и максимальное время цикла (в байтах 10 и 11) могут быть считаны.</li> <li>Для прерываний по ошибке без текущей информации</li> <li>Для прерываний информация о состоянии содержит текущее назначение параметров источника прерывания.</li> <li>Для синхронных ошибок в качестве класса приоритета записывается W#16#7F, если ОВ еще не исполнены, в противном случае - класс приоритета последнего вызова.</li> <li>Если ОВ имеет несколько стартовых событий, и они на момент извлечения информации еще не наступили, то в качестве номера события возвращается W#16#xyzz, где x: класс события, y: не определено, z: наименьший определенный номер группы. В противном случае используется номер последнего появившегося стартового события.</li> </ul>
al 1	1 слово	<p>Идентификатор обработки</p> <p>Бит 0: событие прерывания заблокировано/разблокировано назначением параметра</p> <p>= 0: разблокировано</p> <p>= 1: заблокировано</p> <p>Бит 1: событие прерывания заблокировано с помощью SFC39 "DIS_IRT"</p> <p>= 0: не заблокировано</p> <p>= 1: заблокировано</p> <p>Бит 2 = 1: источник прерывания активен (существует задание на генерирование для прерываний по времени, запущен ОВ прерываний по времени, запущен ОВ прерываний с задержкой, ОВ циклических прерываний: время пошло)</p> <p>Бит 4: ОВ прерываний</p> <p>= 0: не загружен</p> <p>= 1: загружен</p> <p>Бит 5: ОВ прерываний блокирован функцией тестирования и инсталляции</p> <p>= 1: блокирован</p> <p>Бит 6: запись в диагностический буфер</p> <p>= 1: блокирована</p>
al 2	1 слово	<p>Реакция, если ОВ не загружен/ заблокирован</p> <p>Бит 0 = 1: Блокировать источник прерывания</p> <p>Бит 1 = 1: Генерировать ошибку события прерывания</p> <p>Бит 2 = 1: CPU переходит в состояние STOP</p> <p>Бит 3 = 1: Только сбросить прерывание</p>
al 3	2 слова	<p>Отброшено функциями тестирования и инсталляции:</p> <p>Установленный бит № x означает: номер события, который на x больше, чем наименьший номер события соответствующего ОВ, отбрасывается функцией тестирования и инсталляции.</p>

### 31.13 SSL-ID W#16#xy25 - Назначение разделов образа процесса для ОБ

#### Цель

Считывая подписание SSL-ID W#16#xy25, Вы получаете информацию о том, как происходит назначение разделов образа процесса блокам ОБ.

Данный список обеспечивает информацию о:

- разделах образа процесса, которые Вы назначили заданным ОБ для обновления операционной системой
- разделах образа процесса, которые Вы назначили заданным ОБ обработки синхронных прерываний (clock synchronization interrupt) (ОБ 61 ... ОБ 64). Раздел образа процесса здесь обновляется с помощью функций SFC 126 "SYNC\_PI" и 127 "SYNC\_PO".  
Информация о назначениях в системах ведущих DP-устройств и ОБ обработки прерываний синхронизации часов находятся в SSL W#16#xy95.

#### Заголовок

Заголовок подписки SSL-ID W#16#xy25 имеет следующую структуру:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подписки <ul style="list-style-type: none"> <li>• W#16#0025: Назначение разделов образа процесса ОБ в CPU</li> <li>• W#16#0125: Назначение разделов образа процесса соответствующему ОБ Определение ID раздела образа процесса в параметре INDEX.</li> <li>• W#16#0225: Назначение ОБ разделу образа процесса Определение номера ОБ в параметре INDEX. Примечание: ОБ синхронных прерываний (ОБ61...ОБ64) Вы можете назначить нескольким разделам образа процесса.</li> <li>• W#16#0F25: Только информация о заголовке подписки SSL</li> </ul>
INDEX	<ul style="list-style-type: none"> <li>• Для SSL-ID W#16#0025: игнор.</li> <li>• Для SSL-ID W#16#0125: ID раздела образа процесса</li> <li>• Для SSL-ID W#16#0225: номер ОБ</li> <li>• Для SSL-ID W#16#0F25: игнор.</li> </ul>
LENTHDR	W#16#0004: Одна запись данных имеет длину 2 слова (4 байта)
N_DR	Номер записи данных

**Запись данных**

Запись данных фрагмента подсписка SSL-ID W#16#ху25 имеет следующую структуру:

Name	Length	Значение
tra_nr	1 байт	ID раздела образа процесса
tra_use	1 байт	Тип назначения разделов образа процесса и ОБ: <ul style="list-style-type: none"> <li>• Бит 0 = 1: Раздел образа процесса для входов назначается для определенного ОБ для обновления операционной системой.</li> <li>• Бит 1 = 1: Раздел образа процесса для выходов назначается для определенного ОБ для обновления операционной системой.</li> <li>• Бит 2 = 1: Раздел образа процесса для входов назначается для определенного ОБ синхронных прерываний. Он может обновляться в этом ОБ с помощью вызова SFC126 "SYNC_PI".</li> <li>• Бит 3 = 1: Раздел образа процесса для выходов назначается для определенного ОБ синхронных прерываний. Он может обновляться в этом ОБ с помощью вызова SFC126 "SYNC_PO".</li> <li>• Биты 4...7: 0</li> </ul>
ob_nr	1 байт	Номер ОБ
res	1 байт	Резерв

**Фрагменты подсписка**

- Фрагмент подсписка с SSL-ID = W#16#0025:  
Записи данных для всех разделов образа процесса, которые Вы назначили блоку ОБ в своей конфигурации, возвращаются в возрастающем порядке. Значение этого параметра ob\_nr равно нулю для разделов образа процесса, не назначенных никакому ОБ. В этом случае запись данных не возвращается для разделов образа процесса.
- Фрагмент подсписка с SSL-ID = W#16#0125:  
Запись данных возвращается, если Вы назначили адресованный раздел образа процесса блоку ОБ в своей конфигурации. Запись данных не возвращается, если Вы не назначили ОБ.

**Примечание**

Блоку ОБ 1 постоянно назначен раздел образа процесса 0. Таким образом, Вы всегда будете получать запись данных при запросе информации для раздела образа процесса 0.

- Фрагмент подписка с SSL-ID = W#16#0225:  
Записи данных для всех разделов образа процесса, которые Вы назначили вызываемому блоку ОБ. Запись данных не возвращается, если Вы не назначили раздел образа процесса вызываемому блоку ОБ.

---

#### Примечание

Блокам ОБ синхронных прерываний (clock synchronization interrupt) могут быть назначены несколько разделов образа процесса. В этом случае будет возвращаться несколько записей данных.

---

- Фрагмент подписка с SSL-ID = W#16#0F25:  
Возвращается максимальное число (номер) записей данных.

#### Пример значений в записях данных

Вызов параметров SFC51	Возвращаемые переменные	Пояснение
SSL-ID = W#16#0125, INDEX = W#16#0008	tra_nr = B#16#08, tra_use = B#16#03, ob_nr = B#16#15	Запись данных возвращена. Раздел входов/выходов процесса 8 назначены ОБ 21 для обновления операционной системой.
SSL-ID = W#16#0125, INDEX = W#16#0009	–	Запись данных не возвращена. Итог: Раздел образа процесса 9 не назначен никакому ОБ.
SSL-ID = W#16#0225, INDEX = W#16#003D	tra_nr = B#16#0A, tra_use = B#16#C0, ob_nr = B#16#3D  tra_nr = B#16#10, tra_use = B#16#C0, ob_nr = B#16#3D	Две записи данных возвращены. ОБ 61 назначены разделы образа процесса 10 и 16 для входов и выходов. Они могут быть обновлены в ОБ 61 вызовом SFC 126 и SFC 127.
SSL-ID = W#16#0225, INDEX = W#16#0001	tra_nr = B#16#00, tra_use = B#16#03, ob_nr = B#16#01	Одна запись данных возвращена. ОБ 1 назначается образу процесса 0 для входов и выходов, которые могут быть обновлены операционной системой.

## 31.14 SSL-ID W#16#ху32 – Данные о состоянии связи

### Цель

Считывая подсписк с SSL-ID W#16#ху32, то Вы получаете данные о состоянии связи блока.

### Заголовок

Заголовок подсписка SSL-ID W#16#ху32 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подсписка W#16#0132 Данные о состоянии для одного компонента связи CPU (всегда одна запись данных). Компонент связи CPU задается с помощью параметра INDEX. W#16#0232 Данные о состоянии для одного компонента связи (в H-системе в режиме RUN-REDUNDANT [работа с резервированием], возвращается n записей данных, где n – количество резервных CPU в H-системе). Компонент связи CPU задается с помощью параметра INDEX.
INDEX	Компонент связи CPU: <ul style="list-style-type: none"> <li>• для SSL-ID W#16#0132:               <ul style="list-style-type: none"> <li>W#16#0004 уровень защиты CPU, переключатель управления оператором и идентификация версии</li> <li>W#16#0005 диагностика</li> <li>W#16#0008 система времени</li> </ul> </li> <li>• для SSL-ID W#16#0232:               <ul style="list-style-type: none"> <li>W#16#0004 уровень защиты CPU и параметры настройки управления со стороны оператора</li> </ul> </li> </ul>
LENGTHDR	W#16#00028: одна запись данных имеет длину 20 слов (40 байтов)
N_DR	Количество записей данных

### Запись данных

Запись данных подсписка с SSL-ID W#16#0132 всегда имеет длину 20 слов. Записи данных имеют различное содержание. Содержание зависит от параметра INDEX, иными словами, от компонента связи CPU, которому принадлежит эта запись данных.



### 31.15 Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#0005

#### Содержание

Фрагмент подписка с SSL-ID W#16#0132 и индексом W#16#0005 содержит информацию о состоянии диагностики модуля.

#### Запись данных

Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#0005 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	W#16#0005: диагностика
Erw	1 слово	Расширенный набор функций 0: нет 1: да
Send	1 слово	Автоматическая передача 0: нет 1: да
Moeg	1 слово	Передача диагностических сообщений, определенных пользователем, в настоящее время возможна 0: нет 1: да
Res	16 слов	резерв

## 31.16 Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#0008

### Содержание

Фрагмент подписка с SSL-ID W#16#0132 и индексом W#16#0008 содержит информацию о состоянии системы времени модуля.

### Запись данных

Запись данных фрагмента подписка с SSL-ID W#16#01032 и индексом W#16#0008 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	W#16#0008: состояние системы времени
zykl	1 слово	Время цикла кадров синхронизации
korr	1 слово	Кoeffициент коррекции для времени
clock 0	1 слово	Счетчик рабочего времени 0: время в часах
clock 1	1 слово	Счетчик рабочего времени 1: время в часах
clock 2	1 слово	Счетчик рабочего времени 2: время в часах
clock 3	1 слово	Счетчик рабочего времени 3: время в часах
clock 4	1 слово	Счетчик рабочего времени 4: время в часах
clock 5	1 слово	Счетчик рабочего времени 5: время в часах
clock 6	1 слово	Счетчик рабочего времени 6: время в часах
clock 7	1 слово	Счетчик рабочего времени 7: время в часах
time	4 слова	Текущая дата и время (формат: DATE_AND_TIME)
bSSL_0 ... bSSL_1	2 байта	Счетчик рабочего времени активен (бит =1: счетчик рабочего времени активен)
bSSL_0	1 байт	Бит x: счетчик рабочего времени x, $0 \leq x \leq 7$
bSSL_1	1 байт	Резерв
bszu_0 ... bszu_1	2 байта	Переполнение счетчика рабочего времени (бит = 1: переполнение)
bszu_0	1 байт	Бит x: счетчик рабочего времени x, $0 \leq x \leq 7$
bszu_1	1 байт	Резерв
Status	1 слово	Состояние системы времени (о назначении для битов см. ниже)
Res	3 байта	Резерв
status_valid	1 байт	Подтверждение состояния переменных: W#16#01: состояние подтверждается.

## status (состояние)

Бит	Значение по умолчанию	Описание
15	0	Знак для величины корректировки: 0: положительный 1: отрицательный
14 ... 10	00000	Величина корректировки. Данный параметр позволяет скорректировать базовое время в системе отсчета к местному (локальному) времени: Местное время = базовое время ± величина корректировки * 0,5 часа Данная корректировка позволяет учесть пояса времени (временные зоны) и временные различия, связанные с переходом на летнее (система времени для полного использования светового дня) и зимнее время (стандартное время).
9	0	Резерв
8	0	Резерв
7	0	Бит извещения о корректировке времени. Данный параметр показывает, была ли выполнена корректировка времени (включая переход на летнее/зимнее время): 0: корректировка не выполнена; 1: корректировка выполнена.
6	0	Бит извещения о системе установленного (текущего) времени – индикатор "летнее"/"зимнее" время: 0: зимнее время; 1: летнее время.
5	0	В S7 данный бит не используется.
4 ... 3	00	Параметр разрешения времени. В данном параметре задается величина разрешения передаваемого времени: 00: 0,001 с; 01: 0,01 с; 10: 0,1 с; 11: 1 с;
2	0	В S7 данный бит не используется.
1	0	В S7 данный бит не используется.
0	0	Сбой синхронизации. Данный параметр показывает, было ли время передано в фрейме сообщения: 0: сбой синхронизации; 1: синхронизация не нарушена.  Примечание: Проверка данного бита в CPU имеет значение только, если имеет место синхронизация с текущим внешним временем.

## 31.17 Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#000B

### Содержание

Фрагмент подписка с SSL-ID W#16#0132 и индексом W#16#000B содержит информацию о состоянии (status) 32-разрядных счетчиков рабочего времени (run-time meters) с номерами 0 ... 7 модуля.

### Примечание

Фрагмент подписка с SSL-ID W#16#0132 и индексом W#16#0008 отображает состояние счетчиков рабочего времени в 16-разрядном режиме.

Это позволяет использовать программы, созданные для CPU, которые работают с 16-разрядными счетчиками рабочего времени и с фрагментом подписка с SSL-ID W#16#0132 и индексом W#16#0008.

### Запись данных

Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#000B имеет следующую структуру:

Имя	Размер	Значение
Index	1 слово	W#16#000B: Состояние (status) системы времени
bSSL_0	1 байт	Бит x: Состояние (status) счетчика x, $0 \leq x \leq 7$ (Бит = 1: счетчик занят)
bSSL_1	1 байт	Резерв
bszū_0	1 байт	Бит x: Переполнение счетчика x, $0 \leq x \leq 7$ (Бит = 1: переполнение)
bszū_1	1 байт	Резерв
clock 0	2 слова	Счетчик рабочего времени 0: время в часах
clock 1	2 слова	Счетчик рабочего времени 1: время в часах
clock 2	2 слова	Счетчик рабочего времени 2: время в часах
clock 3	2 слова	Счетчик рабочего времени 3: время в часах
clock 4	2 слова	Счетчик рабочего времени 4: время в часах
clock 5	2 слова	Счетчик рабочего времени 5: время в часах
clock 6	2 слова	Счетчик рабочего времени 6: время в часах
clock 7	2 слова	Счетчик рабочего времени 7: время в часах
Res	1 слово	Резерв

## 31.18 Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#000C

### Содержание

Фрагмент подписка с SSL-ID W#16#0132 и индексом W#16#000C содержит информацию о состоянии (status) 32-разрядных счетчиков рабочего времени (run-time meters) с номерами 8 ... 15 модуля.

### Запись данных

Запись данных фрагмента подписка с SSL-ID W#16#0132 и индексом W#16#000C имеет следующую структуру:

Имя	Размер	Значение
Index	1 слово	W#16#000C: Состояние (status) системы времени
BSSL_0	1 байт	Бит x: Состояние (status) счетчика (8+x), $0 \leq x \leq 7$ (Бит = 1: счетчик рабочего времени занят)
BSSL_1	1 байт	Резерв
Bszü_0	1 байт	Бит x: Переполнение счетчика (8+x), $0 \leq x \leq 7$ (Бит = 1: переполнение)
Bszü_1	1 байт	Резерв
Clock 8	2 слова	Счетчик рабочего времени 8: время в часах
Clock 9	2 слова	Счетчик рабочего времени 9: время в часах
Clock 10	2 слова	Счетчик рабочего времени 10: время в часах
Clock 11	2 слова	Счетчик рабочего времени 11: время в часах
Clock 12	2 слова	Счетчик рабочего времени 12: время в часах
Clock 13	2 слова	Счетчик рабочего времени 13: время в часах
Clock 14	2 слова	Счетчик рабочего времени 14: время в часах
clock 15	2 слова	Счетчик рабочего времени 15: время в часах
Res	1 слово	Резерв

### 31.19 Запись данных фрагмента подписка с SSL-ID W#16#0232 и индексом W#16#0004

#### Содержание

Фрагмент подписка с SSL-ID W#16#0232 и индексом W#16#0004 содержит информацию об уровне защиты CPU и параметрах настройки элементов управления со стороны оператора.

В H-системе в режиме RUN-REDUNDANT [работа с резервированием] возвращается одна запись данных на резервный CPU.

#### Запись данных

Запись данных фрагмента подписка с SSL-ID W#16#0232 и индексом W#16#0004 имеет следующую структуру:

Имя	Длина	Значение
Index	1 слово	<ul style="list-style-type: none"> <li>Байт 1: W#16#04: уровень защиты CPU и параметры настройки управления со стороны оператора</li> <li>Байт 0: Стандартный CPU: W#16#00 H CPU: биты 0 ... 2: номер стойки; бит 3: 0 = резервный CPU, 1 = главный CPU; биты 4 ... 7: 1111.</li> </ul>
sch_schal	1 слово	Уровень защиты установлен с помощью переключателя режимов работы (1, 2, 3).
sch_par	1 слово	Уровень защиты установлен в параметрах (0, 1, 2, 3; если 0, то пароля нет, уровень защиты недействителен).
sch_rel	1 слово	Действительный уровень защиты CPU.
bart_sch	1 слово	Положение переключателя режимов (1: RUN, 2: RUN-P, 3: STOP, 4: MRES, 0: неопределенное или не может быть определено).
anl_sch	1 слово	Положение переключателя запуска (1: CRST, 2: WRST, 0: неопределенное, не может быть определено, или переключатель отсутствует).
ken_rel	1 слово	ID для идентификации действительной версии (о – недействительный)
ken_ver1_hw	1 слово	Версия ID 1 для конфигурации оборудования.
ken_ver2_hw	1 слово	Версия ID 2 для конфигурации оборудования.
ken_ver1_awp	1 слово	Версия ID 1 для программы пользователя.
ken_ver2_awp	1 слово	Версия ID 2 для программы пользователя.
Res	8 слов	Резерв

## 31.20 SSL-ID W#16#xy71 – Групповая информация Н CPU

### Цель

Фрагмент частичного списка с SSL- ID W#16#xy71 содержит информацию о текущем состоянии Н-системы.

### Заголовок

Заголовок подписка SSL-ID W#16#xy71 построен следующим образом:

Содержание	Значение
SSL-ID	SSL- ID фрагмента частичного списка: W#16#0071: Информация о текущем состоянии Н-системы W#16#0F71: Информация только заголовка частичного списка SSL
INDEX	W#16#0000
LENGTHDR	W#16#0010: Запись данных имеет длину 8 слов (16 байтов)
N_DR	W#16#0001: Количество записей данных

### Запись данных

Запись данных фрагмента подписка с ID W#16#xy71 имеет следующую структуру:

Содержание	Длина	Значение
Redinf	2 байта	Информация о резервировании W#16#0011: отдельный Н CPU W#16#0012: 1 из 2 Н-систем
Mwstat1	1 байт	Байт состояния 1 Бит 0: Резерв Бит 1: Резерв Бит 2: Резерв Бит 3: Резерв Бит 4: Н-состояние CPU в стойке 0 =0: резервный CPU =1: главный CPU Бит 5: Н-состояние CPU в стойке 1 =0: резервный CPU =1: главный CPU Бит 6: Резерв Бит 7: Резерв

Содержание	Длина	Значение
Mwstat2	1 байт	Байт состояния 2 Бит 0: Состояние соединения синхронизации 01: Синхронизация между CPU 0 и CPU 1 =0: невозможна =1: возможна Бит 1: 0 Бит 2: 0 Бит 3: Резерв Бит 4: =0: CPU не вставлен в стойку 0 =1: CPU вставлен в стойку 0 (в режиме резервирования : бит 4 = 0) Бит 5: =0: CPU не вставлен в стойку 1 =1: CPU вставлен в стойку 1 (в режиме резервирования : бит 5 = 0) Бит 6: Резерв Бит 7: переключение «резервный – главный» после последней депассивации (активации) =0: нет =1: да
Hsfinfo	2 байта	Слово информации для SFC90 "H_CTRL" Бит 0: =0: депассивация не активна =1: депассивация активна Бит 1: =0: обновление резерва разблокировано =1: обновление резерва заблокировано Бит 2: =0: соединение с резервом разблокировано =1: соединение с резервом заблокировано Бит 3: Резерв Бит 4: Резерв Бит 5: Резерв Бит 6: Резерв Бит 7: =1: затребовано модифицирование с обновлением Бит 8: =1: затребовано модифицирование без обновления
Samfehl	2 байта	Резерв
Bz_cpu_0	2 байта	Режим CPU в стойке 0 W#16#0001: STOP (обновление) W#16#0002: STOP (сброс памяти) W#16#0003: STOP (автоинициализация) W#16#0004: STOP (внутренний) W#16#0005: STARTUP (холодный рестарт) W#16#0006: STARTUP (теплый рестарт) W#16#0007: STARTUP (горячий рестарт) W#16#0008: RUN (режим "соло" ) W#16#0009: RUN-R (режим "с резервированием") W#16#000A: HOLD (приостановка) W#16#000B: LINK-UP (соединение) W#16#000C: UPDATE (обновление) W#16#000D: DEFECTIVE (дефектный) W#16#000E: SELFTEST (самотестирование) W#16#000F: NO POWER (нет питания)
Bz_cpu_1	2 байта	Режим CPU в стойке 1 (значения как для bz_cpu_0)
Bz_cpu_2	2 байта	Резерв



Содержание	Длина	Значение
Спу_valid	1 байт	Допустимость переменных bz_спу_0 и bz_спу_1 V#16#01: bz_спу_0 допустимое значение V#16#02: bz_спу_1 допустимое значение V#16#03: bz_спу_0 и bz_спу_1 допустимые значения
reserve	1 байт	Резерв

## 31.21 SSL-ID W#16#ху74 – Состояние светодиодов модуля

### Цель

Если Вы читаете частичный список SSL-ID W#16#ху74, то в случае использования стандартных CPU и в случае H CPU Вы получаете значения состояния светодиодов модулей.

Если H CPU находятся в режиме H без резервирования, то Вы получаете состояние светодиода адресованного CPU. Если H CPU находятся в режиме RUN-REDUNDANT (работа с резервированием), то возвращается состояние светодиодов всех резервных H CPU.

### Заголовок

Заголовок подсписка SSL-ID W#16#ху74 построен следующим образом:

Содержание	Значение
SSL-ID	SSL- ID фрагмента частичного списка W#16#0174 состояние светодиода. Светодиод выбирают с помощью параметра INDEX.
INDEX	ID светодиода (имеет значение только для SSL-ID W#16#0174) W#16#0001: SF (групповая ошибка) W#16#0002: INTF (внутренняя ошибка) W#16#0003: EXTf (внешняя ошибка) W#16#0004: RUN W#16#0005: STOP W#16#0006: FRCE (принудительный) W#16#0007: CRST (холодный рестарт) W#16#0008: BAF (отказ батареи/перегрузка, короткое замыкание напряжения батареи на шину) W#16#0009: USR (определяемый пользователем) W#16#000A: USR1 (определяемый пользователем) W#16#000B: BUS1F (интерфейс ошибки шины 1) W#16#000C: BUS2F (интерфейс ошибки шины 2) W#16#000D: REDF (ошибка резервирования) W#16#000E: MSTR (главный) W#16#000F: RACK0 (стойка номер 0) W#16#0010: RACK1 (стойка номер 1) W#16#0011: RACK2 (стойка номер 2) W#16#0012: IFM1F (интерфейсный модуль 1, ошибка интерфейса) W#16#0013: IFM2F (интерфейсный модуль 2, ошибка интерфейса)
LENGTHDR	W#16#0004: одна запись данных имеет длину 2 слова (4 байта)
N_DR	Число записей данных

**Запись данных**

Запись данных фрагмента подписка с SSL-ID W#16#0074 имеет следующую структуру:

Имя	Длина	Значение
index	1 слово	Байт 0 <ul style="list-style-type: none"><li>• Стандартный CPU: В#16#00</li><li>• Н-CPU:<ul style="list-style-type: none"><li>- Биты с 0 ... 2: номер стойки</li><li>- Бит 3: 0=резервный CPU, 1=главный CPU</li><li>- Биты с 4 ... 7: 1111</li></ul></li></ul> Байт 1: идентификатор светодиода
led_on	1 байт	Состояние светодиода: 0: выключен 1: включен
led_blink	1 байт	Мигающее состояние светодиода: 0: нет мигания 1: мигание с нормальной частотой (2 Гц) 2: мигание с пониженной частотой (0.5 Гц)

## 31.22 SSL-ID W#16#ху75 – Состояние связи между Н-системой и подключенными ведомыми DP-устройствами

### Цель

Если Вы читаете частичный список SSL-ID W#16#ху75, то для CPU в составе Н-систем в режиме с резервированием Вы получаете информацию о состоянии связи между Н-системой и подключенными ведомыми DP-устройствами.

В частичном списке указано, в какой стойке ведущей DP-системы установлен интерфейсный модуль, используемый в текущий момент для связи с подчиненными DP-устройствами.

### Заголовок

Заголовок подсписка SSL-ID W#16#ху75 построен следующим образом:

Содержание	Значение
SSL-ID	SSL- ID фрагмента частичного списка W#16#0C75: состояние связи между Н-системой и подключенными ведомыми DP-устройствами. DP slave-устройство выбирают с помощью параметра INDEX.
INDEX	Адрес диагностируемого (диагностируемых) DP slave-IFM
LENGTHDR	W#16#0010: одна запись данных имеет длину 8 слов (16 байтов)
N_DR	W#16#0001: Число записей данных

**Запись данных**

Запись данных фрагмента подписка с SSL-ID W#16#ху75 имеет следующую структуру:

Имя	Длина	Значение
adr1_bgt0	1 слово	Первая часть адреса подчиненного DP-устройства (интерфейсного модуля), ведущий DP-интерфейсный модуль которого вставлен в стойку 0: ID системы master-DP и номер станции.
adr2_bgt0	1 слово	Вторая часть адреса подчиненного DP-устройства (интерфейсного модуля), ведущий DP-интерфейсный модуль которого вставлен в стойку 0: Слот модуля и слот подмодуля.
adr1_bgt1	1 слово	Первая часть адреса подчиненного DP-устройства (интерфейсного модуля), ведущий DP-интерфейсный модуль которого вставлен в стойку 1: ID системы master-DP и номер станции.
adr2_bgt1	1 слово	Вторая часть адреса подчиненного DP-устройства (интерфейсного модуля), ведущий DP-интерфейсный модуль которого вставлен в стойку 1: Слот модуля и слот подмодуля.
Res	2 слова	Резерв
Logadr	1 слово	Адрес диагностируемого ведомого DP интерфейсного модуля (диагностируемых ведомых DP интерфейсных модулей): <ul style="list-style-type: none"> <li>• Бит 0 ... 14 : логический базовый адрес.</li> <li>• Бит 15 : идентификатор "вход/выход": <ul style="list-style-type: none"> <li>- 0 : вход;</li> <li>- 1 : выход.</li> </ul> </li> </ul>
Slavestatus	1 слово	Состояние связи: <ul style="list-style-type: none"> <li>• Бит 0 = 1: нет доступа к ведомому DP интерфейжному модулю, ведущий DP-интерфейсный модуль которого вставлен в стойку 0.</li> <li>• Бит 1 = 1: нет доступа к ведомому DP интерфейжному модулю, ведущий DP-интерфейсный модуль которого вставлен в стойку 1.</li> <li>• Биты 2 ... 7: Резерв (в каждом бите записан 0).</li> <li>• Бит 8 = 1: Оба канала связи, функционируют нормально; связь в настоящее время реализована через ведущий DP интерфейсный модуль в стойке 0.</li> <li>• Бит 9 = 1: Оба канала связи, функционируют нормально; связь в настоящее время реализована через ведущий DP интерфейсный модуль в стойке 1.</li> <li>• Биты 10 ... 15: Резерв (в каждом бите записан 0).</li> </ul>

### 31.23 SSL-ID W#16#xy90 – состояние связи для всех ведущих DP-систем, зарегистрированных в CPU

#### Цель

Если Вы читаете частичный список SSL-ID W#16#xy90, то Вы получаете информацию о состоянии связи для всех ведущих DP-систем, известных CPU.

#### Заголовок

Заголовок подписка SSL-ID W#16#xy90 построен следующим образом:

Содержание	Значение
SSL-ID	SSL- ID фрагмента частичного списка W#16#0090: Информация обо всех ведущих DP-системах, известных CPU. W#16#0190: Информация об одной ведущей DP-системе. W#16#0F90: Информация только заголовка частичного списка SSL.
INDEX	<ul style="list-style-type: none"> <li>Для фрагмента частичного списка SSL для W#16#0190:  младший байт: B#16#00;  старший байт: ID ведущей DP-системы.</li> <li>Для фрагментов частичного списка SSL для W#16#0090 и W#16#0F90:  W#16#0000.</li> </ul>
LENGTHDR	W#16#000E: одна запись данных имеет длину 7 слов (14 байтов)
N_DR	Число записей данных <ul style="list-style-type: none"> <li>Для фрагмента частичного списка SSL для W#16#0190:  0 ... 1.</li> <li>Для фрагмента частичного списка SSL для W#16#0090:  - для стандартного CPU:  0 ... 14;  - для H-системы:  0 ... 12 (во всех режимах системы, кроме режима с резервированием);  0 ... 2 × 12 (в системе с резервированием).</li> </ul>

**Запись данных**

Запись данных фрагмента подписка с SSL-ID W#16#ху90 имеет следующую структуру:

Имя	Длина	Значение
dp_m_id	1 байт	ID ведущей DP-системы.
rack_dp_m	1 байт	Номер стойки ID ведущей DP-системы: <ul style="list-style-type: none"> <li>• для стандартного CPU: 0;</li> <li>• для H-системы: 0 или 1.</li> </ul>
steckpl_dp_m	1 байт	Слот ведущего DP-устройства или слот CPU (с интегрированным DP-интерфейсом).
subm_dp_m	1 байт	Интерфейсный номер ведущего DP-устройства: <ul style="list-style-type: none"> <li>• для интегрированного DP-интерфейса: <ul style="list-style-type: none"> <li>- 1: X2;</li> <li>- 2: X1;</li> <li>- 3: IF1;</li> <li>- 4: IF2;</li> </ul> </li> <li>• для внешнего DP-интерфейса: 0.</li> </ul>
logadr	1 слово	Логический начальный адрес ведущего DP-устройства.
dp_m_sys_cpu	1 слово	Резерв.
dp_m_sys_dpm	1 слово	Резерв.
dp_m_state	1 байт	Дополнительные характеристики ведущей DP-системы: <p>Бит 0:           Режим DP:</p> <ul style="list-style-type: none"> <li>- 0: S7-совместимость;</li> <li>- 1: DPV1;</li> </ul> <p>Биты 0 ... 6    Резерв.</p> <p>Бит 7:           Тип ведущего DP-устройства:</p> <ul style="list-style-type: none"> <li>- 0: интегрированное устройство;</li> <li>- 1: внешнее устройство.</li> </ul>
reserve	3 байта	Резерв

## 31.24 SSL-ID W#16#ху91 – Информация о состоянии модуля

### Цель

Считывая подписание с SSL-ID W#16#ху91, Вы получаете информацию о состоянии модуля.

### Заголовок

Заголовок подписка SSL-ID W#16#ху91 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подписка
W#16#0091	информация о состоянии модуля для всех вставленных модулей и submodule
W#16#0191	информация о состоянии всех модулей/ стоек с неправильными идентификатором типа
W#16#0291	информация о состоянии всех неисправных модулей
W#16#0391	информация о состоянии всех недоступных модулей
W#16#0591	информация о состоянии всех submodule основного модуля
W#16#0991	информация о состоянии модулей ведущей DP-системы
W#16#0A91	информация о состоянии модулей всех мастер-систем DP
W#16#0C91	информация о состоянии модуля в центральной стойке, подключенного к встроенному интерфейсу DP через логический базовый адрес
W#16#4C91	информация о состоянии модуля, подключенного к внешнему интерфейсу DP через логический базовый адрес
W#16#0D91	информация о состоянии всех модулей в указанной стойке/ станции (DP)
W#16#0E91	информация о состоянии всех сконфигурированных модулей
W#16#0E91	информация о состоянии всех сконфигурированных модулей
W#16#0F91	информация о состоянии всех сконфигурированных модулей
W#16#0F91	число записей данных с информацией о состоянии всех сконфигурированных модулей



Содержание	Значение
INDEX	<ul style="list-style-type: none"> <li>• Для фрагмента подписка с SSL-ID W#16#0C91: <ul style="list-style-type: none"> <li>-S7-400: биты 0 ... 14: логический базовый адрес модуля бит 15: 0 = вход, 1 = выход</li> <li>-S7-300: начальный адрес модуля</li> </ul> </li> <li>• Для фрагмента подписка с SSL-ID W#16#4C91 (только S7-400): <ul style="list-style-type: none"> <li>биты 0 ... 14: логический базовый адрес модуля</li> <li>бит 15: 0 = вход, 1 = выход</li> </ul> </li> <li>• Для фрагмента подписка с SSL-ID W#16#0091, W#16#0191, W#16#0291, W#16#0391, W#16#0491, W#16#0591, W#16#0A91, W#16#0E91, W#16#0F91: INDEX не имеет значения, все модули (в стойке и в децентрализованной периферии)</li> <li>• Для фрагмента подписка с SSL-ID W#16#0991 и W#16#0D91: <ul style="list-style-type: none"> <li>W#16#00xx все модули и submodule стойки (xx содержит номер стойки)</li> <li>W#16#xx00 все модули ведущей DP-системы (xx содержит идентификатор ведущей DP-системы)</li> <li>W#16#xxуу все модули DP-станции (xx содержит идентификатор ведущей DP-системы, уу - номер станции)</li> </ul> </li> </ul>
LENGTHDR	W#16#0010: одна запись данных имеет длину 8 слов (16 байтов)
N_DR	Количество записей данных. В зависимости от программы число записей, перемещенных в SFC 51 может быть меньше.

### Записи данных

В случае W#16#0091, W#16#0191 и W#16#0F91 две дополнительные записи приходятся на каждую стойку:

- запись для источника питания, если он присутствует или планируется;
- запись для стойки;

Последовательность записей для централизованной структуры:

ИП, слот1, слот2, слот3, ..., слот18, стойка;

Запись данных подписка с ID W#16#ху91 имеет следующую структуру:

Имя	Длина	Значение
adr1	1 слово	Номер стойки (идентификатор ведущей DP-системы с DP) (географический адрес)
adr2	1 слово	Слот модуля и слот submodule
logadr	1 слово	Первый назначенный логический адрес входа/выхода (базовый адрес)
solltyp	1 слово	Резерв
isttyp	1 слово	Резерв
alarm	1 слово	Резерв (00xx=CPU № 1...4)
eastat	1 слово	Состояние ввода/вывода Бит 0 = 1: модуль неисправен (обнаружено через диагностическое прерывание) Бит 1 = 1: модуль существует Бит 2 = 1: модуля нет (обнаружено по ошибке доступа) Бит 5 = 1: модуль может быть основным модулем для submodule Бит 6 = 1: Резерв для S7-400 Бит 7 = 1: модуль в локальном сегменте шины Биты 8 ... 15: идентификатор данных для логического адреса (вход: V#16#B4, выход: V#16#B5, внешний интерфейс DP: V#16#FF)
ber_bgbr	1 слово	Идентификатор области/ширина модуля Биты 0 ... 2 : ширина модуля Бит 3: резерв Биты 4 ... 6 : идентификатор области 0 = S7-400 1 = S7-300 2 = область ET 3 = область P 4 = область Q 5 = область IM3 6 = область IM4 Бит 7: резерв

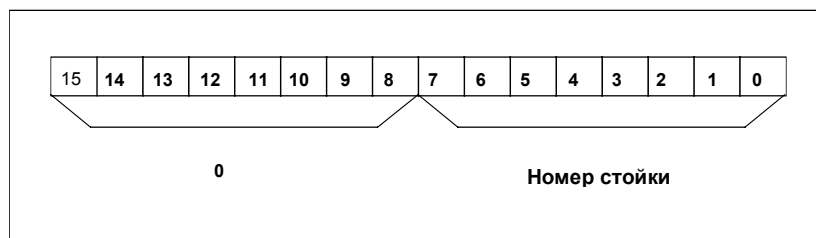
В некоторых модулях в записи обозначаются следующие значения:

<b>Имя</b>	<b>ИП</b> (Только для S7-400)	<b>CPU</b>	<b>IFM-CPU</b> (Для S7-300)	<b>Стойка</b> (Только для S7-400)
adr1	Номер стойки	Стандартная информация в соответствии с данным выше описанием	Стандартная информация в соответствии с данным выше описанием	Номер стойки
adr2	W#16#01FF	W#16#0200 или W#16#0200 ... W#16#1800	W#16#0200	W#16#00FF
logadr	W#16#0000	W#16#7FFF	W#16#007C	W#16#0000
solltyp	Стандартная информация в соответствии с данным выше описанием	W#16#00C0 или W#16#0081 или W#16#0082	W#16#00C0	Стандартная информация в соответствии с данным выше описанием
eastat	W#16#0000	Стандартная информация в соответствии с данным выше описанием	Стандартная информация в соответствии с данным выше описанием	W#16#0000
ber_bgr	W#16#0000	W#16#0011 или W#16#0001 или W#16#0002	W#16#0011	W#16#0000

**adr1**

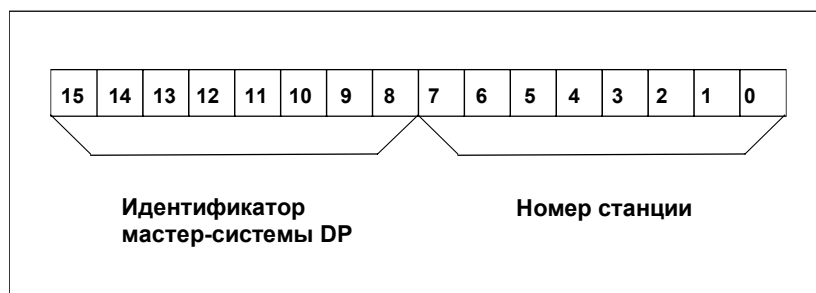
Параметр adr1 содержит следующую информацию:

- при центральной установке – номер стойки.



*Биты параметра adr1 при центральной установке*

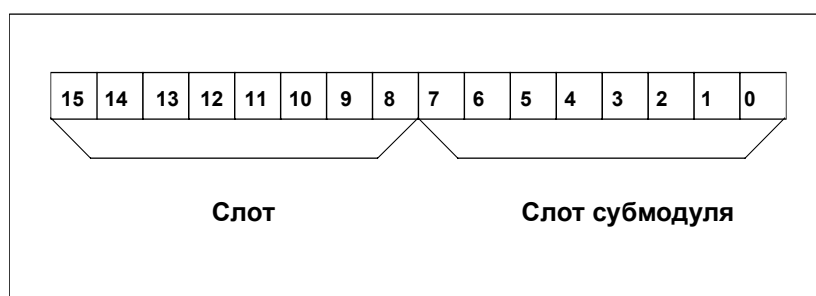
- в случае децентрализованной конфигурации  
идентификатор ведущей DP-системы  
номер станции.



*Биты параметра adr1 в децентрализованной конфигурации.*

**adr2**

Параметр adr2 содержит слот модуля и слот submodule.



*Биты параметра adr2*

---

**Информация для мультипроцессорных систем**

Только частичные списки дают информацию относительно модулей, которые назначены CPU. Следовательно, для режима обработки данных в многопроцессорной системе Вы должны произвести выборку всех CPU, чтобы получить данные обо всех подключенных модулях.

---

## 31.25 SSL-ID W#16#xy92 – Информация о состоянии стойки/станции

### Цель

Считывая подписание с SSL-ID W#16#xy92, Вы получаете информацию об ожидаемой и текущей конфигурации аппаратуры стоек централизованной структуры и станций ведущей DP-системы.

### Считывание SSL с помощью SFC51 „RDSYSST“ с использованием S7-400 CPU

Если Вы считываете частичный список с помощью SFC51, Вы должны позаботиться о том, чтобы параметры SSL-ID и INDEX функции SFC51 соответствовали друг другу.

SSL-ID	INDEX
W#16#0092 или W#16#0292 или W#16#0692	ID ведущей DP-системы, которая подключена посредством интегрированного DP-включения.
W#16#4092 или W#16#4292 или W#16#4692	ID ведущей DP-системы, которая подключена посредством внешнего DP-включения.

**Заголовок**

Заголовок подписка SSL-ID W#16#ху92 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подписка: W#16#0092: ожидаемое состояние центральных стоек/станций ведущей DP-системы, подключенных через встроенный интерфейс DP; W#16#4092: ожидаемое состояние станций ведущей DP-системы, подключенных через внешний интерфейс DP; W#16#0292: фактическое состояние центральных стоек/станций ведущей DP-системы, подключенных через встроенный интерфейс DP; W#16#0392: фактическое состояние буферной батареи, резервирующей электропитание стойки / станции CPU, если по крайней мере одна батарея вышла из строя; W#16#0492: состояние системы буферного батарейного питания в целом, резервирующего электропитание всех стоек / станций CPU; W#16#0592: состояние источника питания 24 В для всех стоек/станций CPU; W#16#4292: фактическое состояние ведущей DP-системы, подключенных через внешний интерфейс DP; W#16#0692: нормальное состояние стоек расширения в центральной конфигурации/ станций мастер-системы DP, подключенных через встроенный интерфейс DP; W#16#4692: нормальное состояние станций ведущей DP-системы, подключенных через внешний интерфейс DP.
INDEX	0/ идентификатор ведущей DP-системы
LENGTHDR	W#16#0010: одна запись данных имеет длину 8 слов (16 байтов)
N_DR	Количество записей данных

**Запись данных**

Запись данных подписка с ID W#16#ху92 имеет следующую структуру:

Содержание	Длина	Значение
status_0 ... status_15	16 байтов	<p>Состояние стойки/станции или состояние поддержки</p> <p>W#16#0092:   Бит=0: стойка/станция не сконфигурирована                   Бит=1: стойка/станция сконфигурирована</p> <p>W#16#4092   Бит=0: станция не сконфигурирована                   Бит=1: станция сконфигурирована</p> <p>W#16#0192:   Бит=0: станция не сконфигурирована или не сконфигурирована и активирована                   Бит=1: станция сконфигурирована и активирована</p> <p>W#16#0292:   Бит=0: стойка/станция неисправна или не сконфигурирована                   Бит=1: стойка/станция существует и исправна</p> <p>W#16#4292:   Бит=0: станция неисправна или не сконфигурирована                   Бит=1: станция существует, исправна и активирована.</p> <p>W#16#0692:   Бит=0: все модули стойки расширения /станции существуют, доступны, ошибок нет и станция активирована.                   Бит=1: по крайней мере, 1 модуль стойки расширения/ станции не в порядке или станция деактивирована.</p> <p>W#16#4692:   Бит=0: все модули станции существуют, доступны, ошибок нет, и станция активирована.                   Бит=1: по крайней мере, 1 модуль станции не в порядке или станция деактивирована.</p>
status_0	1 байт	<p>Бит 0:           центральная стойка (INDEX = 0) или станция 1 (INDEX ≠ 0)</p> <p>Бит 1:           1. Стойка расширения или станция 2                   :                   : Бит 7:           7. Стойка расширения или станция 8</p>
status_1	1 байт	<p>Бит 0:           8. Стойка расширения или станция 9                   :                   : Бит 7:           15. Стойка расширения или станция 16</p>



Содержание	Длина	Значение	
status_2	1 байт	Бит 0:	16. Стойка расширения или станция 17 : : Бит 5: 21. Стойка расширения или станция 22 Бит 6: 0 или станция 23 Бит 7: 0 или станция 24
status_3	1 байт	Бит 0:	0 или станция 25 : : Бит 5: 0 или станция 30 Бит 6: стойка расширения (область SIMATIC S5) или станция 31 Бит 7: 0 или станция 32
status_4	1 байт	Бит 0:	0 или станция 33 : : Бит 7: 0 или станция 40
:			
status_15	1 байт	Бит 0:	0 или станция 121 : : Бит 7: 0 или станция 128

## 31.26 SSL-ID W#16#ху95 - расширенная информация о состоянии систем ведущих DP-устройств

### Цель

Подсписок с SSL-ID W#16#ху95 обеспечивает пользователя расширенной информацией о состоянии систем ведущих DP-устройств, известных CPU. По сравнению с подсписком с SSL-ID W#16#ху90 данный список содержит дополнительную информацию по синхронизации часов (clock synchronism) системы ведущего DP-устройства.

### Заголовок

Заголовок подсписка SSL-ID W#16#ху95 имеет следующую структуру:

Содержание	Значение	
SSL-ID	SSL-ID фрагмента подсписка	
	W#16#0195:	Расширенная информация о системе ведущего DP-устройства
	W#16#0F95:	Только информация о заголовке подсписка SSL
INDEX	<ul style="list-style-type: none"> <li>• Для фрагмента подсписка с SSL-ID W#16#0195:</li> <li>• Младший байт: W#16#00</li> <li>• Старший байт: ID системы ведущего DP-устройства</li> <li>• Для фрагмента подсписка с SSL-ID W#16#0F95:</li> <li>• W#16#0000</li> </ul>	
LENTHDR	W#16#0028:	Одна запись данных имеет размер 20 слов (40 байт)
N_DR	Номер записи данных: для фрагмента подсписка с SSL-ID W#16#0195: 0 или 1	

**Запись данных**

Запись данных подписка с ID W#16#ху95 имеет следующую структуру:

Имя	Размер	Значение
dp_m_id	1 байт	ID системы ведущего DP-устройства
rack_dp_m	1 байт	Номер стойки модуля системы ведущего DP-устройства <ul style="list-style-type: none"> <li>• для стандартного CPU: 0;</li> <li>• для H-системы: 0 или 1</li> </ul>
steckpl_dp_m	1 байт	Слот ведущего DP-устройства или слот CPU (с интегрированным DP-интерфейсом).
subm_dp_m	1 байт	<ul style="list-style-type: none"> <li>• для интегрированного DP-интерфейса: ID интерфейса ведущего DP-устройства: <ul style="list-style-type: none"> <li>- 1: X2</li> <li>- 2: X1</li> <li>- 3: IF1</li> <li>- 4: IF2</li> </ul> </li> <li>• для внешнего DP-интерфейса: 0.</li> </ul>
logadr	2 байта	Логический начальный адрес ведущего DP-устройства
dp_m_sys_cpu	2 байта	Резерв
dp_m_sys_dpm	2 байта	Резерв
dp_m_state	1 байт	Дополнительные характеристики системы ведущего DP-устройства: Бит 0: Режим DP: <ul style="list-style-type: none"> <li>• 0: S7-совместимость (S7 compatible);</li> <li>• 1: DPV1</li> </ul> Бит 1: DP-цикл <ul style="list-style-type: none"> <li>• 0: неравномерный (неэквидистантный)</li> <li>• 1: равномерный (эквидистантный)</li> </ul> Биты 2...6: Резерв Бит 7: Тип ведущего DP-устройства: <ul style="list-style-type: none"> <li>• 0: интегрированное ведущее DP-устройство;</li> <li>• 1: внешнее ведущее DP-устройство.</li> </ul>
reserve	3 байта	Резерв
tsal_ob	1 байт	Назначенный OB прерывания синхронизации часов (только если DP-цикл является равномерным (эквидистантным))
reserve	1 байт	Резерв
baudrate	4 байта	Скорость передачи системы ведущего DP-устройства (hex формат)
dp_iso_takt	4 байта	Период для равномерного (эквидистантного) DP-цикла (в мкс)
reserve	16 байтов	Резерв

**Информация по многопроцессорному режиму (только для S7-400)**

Все подписки дают информацию только для модулей, которые назначены CPU. Следовательно, в данном режиме Вы должны опросить все CPU, чтобы получать данные от всех подключенных модулей.

## 31.27 SSL-ID W#16#хуА0 – Диагностический буфер

### Цель

Считывая подсписк с SSL-ID W#16#хуА0, Вы получаете записи из диагностического буфера модуля.

### Заголовок

Заголовок подсписка SSL-ID W#16#хуА0 построен следующим образом:

Содержание	Значение
SSL-ID	SSL-ID фрагмента подсписка: W#16#00A0: все записи, возможные в текущем режиме W#16#01A0: самые новые записи; количество самых новых задается через параметр INDEX. Если число сообщений в диагностическом буфере меньше, чем сконфигурированное максимальное число сообщений, SFC51 может выдать недопустимые значения, используя фрагмент частичного списка. Следовательно, Вы должны избежать потери электропитания, которое не резервируется!  W#16#0FA0: только информация заголовка подсписка
INDEX	Только для SSL-ID W#16#01A0: Количество самых новых записей
LENGTHDR	W#16#0014: запись данных имеет длину 10 слов (20 байтов)
N_DR	Количество записей данных

### Запись данных

Запись данных подсписка с SSL-ID W#16#хуА0 имеет следующую структуру:

Имя	Длина	Значение
ID	1 слово	Идентификатор события
info	5 слов	Информация о событии и его последствиях
time	4 слова	Отметка времени события

### Диагностический буфер

Дополнительную информацию к событиям из диагностического буфера Вы можете получить с помощью STEP 7.

## 31.28 SSL-ID W#16#00B1 – Диагностическая информация модуля

### Цель

Считывая подписание с SSL-ID W#16#00B1, Вы получаете первые 4 диагностических байта модуля, обладающего встроенными средствами диагностики.

### Заголовок

Заголовок подписка SSL-ID W#16#00B1 построен следующим образом:

Содержание	Значение
SSL-ID	W#16#00B1
INDEX	Бит 0 ... бит 14: логический базовый адрес Бит 15: 0 = вход, 1 = выход
LENGTHDR	W#16#0004: запись данных имеет длину 2 слова (4 байта)
N_DR	1

### Запись данных

Запись данных подписка с SSL-ID W#16#00B1 имеет следующую структуру:

Имя	Длина	Значение
byte1	1 байт	Бит 0: модуль неисправен/ОК (идентификатор групповой ошибки) Бит 1: внутренняя ошибка Бит 2: внешняя ошибка Бит 3: имеется ошибка канала Бит 4: отсутствует внешнее вспомогательное напряжение Бит 5: отсутствует передний штепсельный разъем Бит 6: модулю не назначены параметры Бит 7: неправильные параметры в модуле
byte2	1 байт	Бит 0 ... Бит 3: класс модуля (CPU, FM, CP, IM, SM, ...) Бит 4: имеется информация канала Бит 5: имеется информация пользователя Бит 6: диагностическое прерывание из-за замены Бит 7: резерв (инициализирован 0)

Имя	Длина	Значение
byte3	1 байт	Бит 0: модуль пользователя некорректен/отсутствует Бит 1: нарушение связи Бит 2: рабочий режим RUN/STOP (0 = RUN, 1 = STOP) Бит 3: сработал контроль времени Бит 4: отказал внутренний источник питания модуля Бит 5: разрядилась батарея (BFS) Бит 6: отказ всей буферизации Бит 7: резерв (инициализирован 0)
byte4	1 байт	Бит 0: отказ стойки расширения (обнаружен с помощью IM) Бит 1: отказ процессора Бит 2: ошибка СППЗУ Бит 3: ошибка ОЗУ Бит 4: ошибка АЦП/ЦПУ Бит 5: сгорел предохранитель Бит 6: потеряно аппаратное прерывание Бит 7: резерв (инициализирован 0)

## 31.29 SSL-ID W#16#00B2 – Диагностическая запись данных 1 с географическим адресом

### Цель

Считывая подписание с SSL-ID W#16#00B2, Вы получаете диагностическую 1 запись данных модуля в центральной стойке (не для DP и не для субмодулей). Модуль задается номером стойки и слота.

### Заголовок

Заголовок подписка SSL-ID W#16#00B2 построен следующим образом:

Содержание	Значение
SSL-ID	W#16#00B2
INDEX	W#16#ххуу:      хх содержит номер стойки уу содержит номер слота
LENGTHDR	Длина записи данных зависит от модуля.
N_DR	1

### Запись данных

Размер записи данных подписка с SSL-ID W#16#00B2 и ее содержание зависят от конкретного модуля. За дополнительной информацией обратитесь к [/70/](#), [/101/](#) и к руководству, описывающему соответствующий модуль.

### 31.30 SSL-ID W#16#00B3 – Диагностические данные модуля с логическим базовым адресом

#### Цель

Считывая подсписк с SSL-ID W#16#00B3, Вы получаете все диагностические данные модуля. Вы можете получить эту информацию также для DP и submodule. Модуль выбирается с помощью его логического базового адреса.

#### Заголовок

Заголовок подсписка SSL-ID W#16#00B3 построен следующим образом:

Содержание	Значение
SSL-ID	W#16#00B3
INDEX	Бит 0 ... Бит 14: логический базовый адрес Бит 15: 0 =вход, 1 = выход
LENGTHDR	Длина записи данных зависит от модуля.
N_DR	1

#### Запись данных

Размер записи данных подсписка с SSL-ID W#16#00B3 и ее содержание зависит от конкретного модуля.

За дополнительной информацией обратитесь к [/70/](#), [/101/](#) и к руководству, описывающему соответствующий модуль.

---

#### Примечание

С помощью SFC51 Вы должны считать подсписк SSL-ID W#16#16#00B3 только не из OB82.

---



### 31.31 SSL-ID W#16#00B4 – Диагностические данные ведомых DP-устройств

#### Цель

Считывая подписание с SSL-ID W#16#00B4, Вы получаете диагностические данные ведомого DP-устройства. Эти диагностические данные структурированы в соответствии с EN 50 170 Volume 2, PROFIBUS. Модуль выбирается с помощью сконфигурированного вами диагностического адреса.

#### Заголовок

Заголовок подписка SSL-ID W#16#00B4 построен следующим образом:

Содержание	Значение
SSL-ID	W#16#00B4
INDEX	Сконфигурированный диагностический адрес ведомые DP-устройства
LENGTHHDR	Длина записи данных. Максимальная длина составляет 240 байтов. Для стандартных slave-устройств, имеющих длину диагностических данных больше 240 байтов, но не более 244 байтов, считываются первые 240 байтов, и в данных устанавливается бит переполнения.
N_DR	1

#### Запись данных

Запись данных подписка с SSL-ID W#16#00B4 имеет следующую структуру:

Имя	Длина	Значение
status1	1 байт	Состояние станции 1
status2	1 байт	Состояние станции 2
status3	1 байт	Состояние станции 3
stat_nr	1 байт	Номер главной станции
ken_hi	1 байт	Идентификатор изготовителя (старший байт)
ken_lo	1 байт	Идентификатор изготовителя (младший байт)
....	....	Дополнительная диагностика, специфическая для конкретного slave-устройства



## **32 События**

## 32.1 События и идентификаторы (ID) событий

### Событие

Все события в программируемых логических контроллерах SIMATIC S7 нумеруются. Это позволяет поставить в соответствие событию текст сообщения.

### Идентификатор (ID) события

Каждому событию соответствует идентификатор (ID) события. ID построен следующим образом:



Структура ID события.

### Класс события

События разделены на классы следующим образом:

Номер	Класс события
1	События, связанные со стандартными ОВ
2	Синхронные ошибки
3	Асинхронные ошибки
4	Переходы из одного рабочего режима в другой
5	События этапа выполнения
6	Коммуникационные события
7	События в отказоустойчивых системах
8	Стандартизованные диагностические данные в модулях
9	События, предопределяемые пользователем
A, B	Свободно определяемые события
C, D, E	Резерв
F	События для модулей, отличных от CPU (например, CP, FM)

**Идентификатор**

Идентификатор используется, чтобы различать типы событий. Эти четыре бита имеют следующие значения:

№ бита в ID события	Значение	
8	= 0	уходящее событие
	= 1	наступающее событие
9	= 1	запись в диагностический буфер
10	= 1	внутренняя ошибка
11	= 1	внешняя ошибка

## 32.2 Класс событий 1 – События, связанные со стандартными ОВ

ID события	Событие
W#16#1164	Прерывание синхронного цикла 1
W#16#1165	Прерывание синхронного цикла 2
W#16#1166	Прерывание синхронного цикла 3
W#16#1167	Прерывание синхронного цикла 4
W#16#1381	Запрос на ручной теплый рестарт
W#16#1382	Запрос на автоматический теплый рестарт
W#16#1383	Запрос на ручной горячий рестарт
W#16#1384	Запрос на автоматический горячий рестарт
W#16#1385	Запрос на ручной холодный рестарт
W#16#1386	Запрос на автоматический холодный рестарт
W#16#1387	Основной CPU: запрос на ручной холодный рестарт
W#16#1388	Основной CPU: запрос на автоматический холодный рестарт
W#16#138A	Основной CPU: запрос на ручной теплый рестарт
W#16#138B	Основной CPU: запрос на автоматический теплый рестарт
W#16#138C	Резервный CPU: запрос на ручной горячий рестарт
W#16#138D	Резервный CPU: запрос на автоматический горячий рестарт

### 32.3 Класс событий 2 – Синхронные ошибки

ID события	Событие	ОВ
W#16#2521	Ошибка BCD - преобразования	OB121
W#16#2522	Ошибка длины области при чтении	
W#16#2523	Ошибка длины области при записи	
W#16#2524	Ошибка области при чтении	
W#16#2525	Ошибка области при записи	
W#16#2526	Ошибка номера таймера	
W#16#2527	Ошибка номера счетчика	
W#16#2528	Ошибка выравнивания при чтении	
W#16#2529	Ошибка выравнивания при записи	
W#16#2530	Ошибка записи при обращении к DB	
W#16#2531	Ошибка записи при обращении к DI	
W#16#2532	Ошибка номера блока при открытии DB	
W#16#2533	Ошибка номера блока при открытии DI	
W#16#2534	Ошибка номера блока при вызове FC	
W#16#2535	Ошибка номера блока при вызове FB	
W#16#253A	DB не загружен	
W#16#253C	FC не загружен	
W#16#253D	SFC не загружен	
W#16#253E	FB не загружен	
W#16#253F	SFB не загружен	
W#16#2942	Ошибка доступа к периферии, чтение	OB122
W#16#2943	Ошибка доступа к периферии, запись	
W#16#2944	Ошибка доступа к периферии при n-ном обращении для чтения (n>1)	
W#16#2945	Ошибка доступа к периферии при n-ном обращении для записи (n>1)	

### 32.4 Класс событий 3 – Асинхронные ошибки

ID события	Событие	ОВ
W#16#3501	Превышение длительности цикла.	OB80
W#16#3502	Ошибка запроса пользовательского интерфейса (ОВ или FRB)	
W#16#3503	Слишком большая задержка при обработке класса приоритета	-
W#16#3505	Прерывание(я) по времени, пропущенное(ые) из-за новой установки часов	OB80
W#16#3506	Прерывание(я) по времени, пропущенное(ые) при переходе в RUN после HOLD	
W#16#3507	Многочисленные ошибки при запросе ОВ, вызванные переполнением буфера стартовой информации	
W#16#3508	Ошибка установки прерывания синхронного цикла	
W#16#3921/3821	BATTF: выход из строя по крайней мере одной буферной батареи в центральной стойке/ проблема устранена Примечание: Событие, изменяющее состояние происходит, только если одна из резервных батарей отказывает (если имеются таковые). Если другая резервная батарея также отказывает, событие не наступает.	OB81
W#16#3922/3822	BAF: сбой резервного источника напряжения в центральной стойке / проблема устранена	
W#16#3923/3823	Выход из строя источника 24 В в центральной стойке / проблема устранена	
W#16#3925/3825	BATTF: выход из строя по крайней мере одной буферной батареи резервной центральной стойки/ проблема устранена	
W#16#3926/3826	BAF: сбой резервного источника напряжения в резервной центральной стойке / проблема устранена	
W#16#3917/3827	Выход из строя источника 24 в резервной центральной стойке / проблема устранена	
W#16#3931/3831	BATTF: выход из строя по крайней мере одной буферной батареи в стойке расширения/ проблема устранена	
W#16#3932/3832	BAF: сбой резервного источника напряжения в стойке расширения/ проблема устранена	
W#16#3933/3833	Выход из строя источника 24 В по крайней мере в одной стойке расширения / проблема устранена	
W#16#3942	Модуль неисправен	
W#16#3842	Модуль исправен	
W#16#3861	Модуль/интерфейсный модуль установлен, тип модуля верен	OB83
W#16#3961	Модуль/интерфейсный модуль удален, недоступен	
W#16#3863	Модуль/интерфейсный модуль установлен, но неверен тип модуля	
W#16#3864	Модуль/интерфейсный модуль установлен, но поврежден (не читается идентификатор типа)	



ID события	Событие	ОВ	
W#16#3866	Модуль установлен, но неверно назначены параметры модуля	ОВ83	
W#16#3966	Модуль может быть снова доступен, устранен сбой напряжения нагрузки.		
W#16#3865	Модуль не может быть адресован, сбой напряжения нагрузки.		
W#16#3884	Интерфейсный модуль вставлен		
W#16#3984	Интерфейсный модуль удален		
W#16#3981	Ошибка интерфейса, наступающее событие	ОВ84	
W#16#3881	Ошибка интерфейса, уходящее событие		
W#16#35A1	Пользовательский интерфейс (ОВ или FRB) не найден	ОВ85	
W#16#35A2	ОВ не загружен (запущен посредством SFC или операционной системы в соответствии с конфигурацией)		
W#16#35A3	Ошибка при обращении к блоку операционной системы		
W#16#39B1	Ошибка доступа к периферии при обновлении таблицы входов образа процесса	ОВ85	
W#16#39B2	Ошибка доступа к периферии при передаче образа процесса в модули вывода		
W#16#39B3/38B3	Ошибка доступа к периферии при обновлении таблицы входов образа процесса		
W#16#39B4/38B4	Ошибка доступа к периферии при передаче образа процесса в модули вывода		
W#16#38C1	Выход из строя стойки расширения (1 - 21), уходящее событие	ОВ86	
W#16#39C1	Выход из строя стойки расширения (1 - 21), наступающее событие		
W#16#38C2	Стойка расширения восстановлена, но с расхождением между заданной и фактической конфигурацией		
W#16#39C3	Децентрализованная периферия: отказ ведущей системы, наступающее событие		
W#16#39C4	Децентрализованная периферия: отказ станции, наступающее событие		
W#16#38C4	Децентрализованная периферия: отказ станции, уходящее событие		
W#16#39C5	Децентрализованная периферия: отказ станции, наступающее событие		
W#16#38C5	Децентрализованная периферия: отказ станции, уходящее событие		
W#16#38C6	Стойка расширения восстановлена, но ошибка(и) в назначениях параметров модуля		
W#16#38C7	DP: станция восстановлена, но ошибка(и) в назначениях параметров модуля		
W#16#38C8	DP: станция восстановлена, но несовпадение между заданной и фактической конфигурацией		
W#16#35D2	Передача диагностических записей в данное время невозможна		ОВ87
W#16#35D3	Фреймы синхронизации не могут быть посланы		

<b>ID события</b>	<b>Событие</b>	<b>ОВ</b>
W#16#35D4	Недопустимый скачок времени в результате синхронизации	ОВ 87
W#16#35D5	Ошибка при приеме времени синхронизации	
W#16#35E1	Неверный идентификатор фрейма в GD	
W#16#35E2	Статус пакета GD не может быть занесен в DB	
W#16#35E3	Ошибка длины фрейма в GD	
W#16#35E4	Принят недопустимый номер пакета GD	
W#16#35E5	Ошибка при обращении к DB в коммуникационных SFB для конфигурирования соединений S7	
W#16#35E6	Общий статус GD не может быть занесен в DB	

### 32.5 Класс событий 4 – События, связанные с режимом STOP, и другие изменения режима работы

ID события	Событие
W#16#4300	Включение резервного питания
W#16#4301	Переход из режима STOP в режим STARTUP (запуск)
W#16#4302	Переход из режима STARTUP (запуск) в RUN (выполнение)
W#16#4303	Режим STOP в результате перевода в STOP переключателя режимов работы
W#16#4304	STOP в результате команды STOP от PG или через SFB20 "STOP"
W#16#4305	HOLD: достигнута точка останова
W#16#4306	HOLD: точка останова покинута
W#16#4307	Запуск сброса памяти командой из PG
W#16#4308	Запуск сброса памяти от переключателя режимов
W#16#4309	Автоматический запуск сброса памяти (небуферизованное включение питания)
W#16#430A	HOLD покинут, переход в режим STOP
W#16#430D	STOP, вызванный другим CPU в мультипроцессорном режиме
W#16#430E	Сброс памяти выполнен
W#16#430F	STOP в модуле из-за перехода в STOP в CPU
W#16#4510	STOP, вызванный нарушением диапазона данных CPU
W#16#4520	DEFECTIVE: STOP невозможен
W#16#4521	DEFECTIVE: отказ процессора обработки команд
W#16#4522	DEFECTIVE: отказ микросхемы часов
W#16#4523	DEFECTIVE: отказ генератора тактовых импульсов
W#16#4524	DEFECTIVE: отказ функции обновления таймера
W#16#4525	DEFECTIVE: отказ многопроцессорной синхронизации
W#16#4926	DEFECTIVE: отказ контроля времени при обращении к периферии
W#16#4527	DEFECTIVE: отказ контроля обращения к периферии
W#16#4528	DEFECTIVE: отказ контроля длительности цикла
W#16#4530	DEFECTIVE: ошибка тестирования внутренней памяти
W#16#4931	DEFECTIVE: ошибка тестирования памяти submodule
W#16#4532	DEFECTIVE: отказ ресурсов ядра
W#16#4933	DEFECTIVE: ошибка контрольной суммы
W#16#4934	DEFECTIVE: память недоступна
W#16#4935	DEFECTIVE: отменено сторожевой схемой/ ненормальные состояния процессора
W#16#4536	DEFECTIVE: неисправен переключатель режимов работы
W#16#4540	STOP: расширение внутренней рабочей памяти имеет пробелы. Первое расширение памяти слишком мало или отсутствует.

ID события	Событие
W#16#4541	STOP, вызванный системой классов приоритета
W#16#4542	STOP, вызванный системой управления объектом
W#16#4543	STOP, вызванный функциями тестирования
W#16#4544	STOP, вызванный диагностической системой
W#16#4545	STOP, вызванный системой связи
W#16#4546	STOP, вызванный управлением памятью CPU
W#16#4547	STOP, вызванный управлением образом процесса
W#16#4548	STOP, вызванный управлением периферией
W#16#4949	STOP, вызванный непрерывным аппаратным прерыванием
W#16#454A	STOP, вызванный конфигурированием, отмененный OB был загружен при полном рестарте
W#16#494D	STOP, вызванный ошибкой ввода/вывода
W#16#494E	STOP, вызванный сбоем питания
W#16#494F	STOP, вызванный ошибкой конфигурации
W#16#4550	DEFECTIVE: внутренняя системная ошибка
W#16#4555	Повторный запуск невозможен, истекло время контроля
W#16#4556	STOP: запрос на сброс памяти от системы связи
W#16#4357	Запущено время контроля модуля
W#16#4358	Все модули готовы к работе
W#16#4959	STOP: не все модули готовы к работе
W#16#4562	STOP, вызванный ошибкой программирования (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#4563	STOP, вызванный ошибкой доступа к периферии (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#4567	STOP, вызванный H-событием
W#16#4568	STOP, вызванный ошибкой времени (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#456A	STOP, вызванный диагностическим прерыванием (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#456B	STOP, вызванный снятием/установкой модуля (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#456C	STOP, вызванный аппаратной ошибкой CPU (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#456D	STOP, вызванный ошибкой исполнения программы (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#456E	STOP, вызванный ошибкой связи (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#456F	STOP, вызванный неисправностью стойки (OB не загружен или его запуск невозможен или отсутствует FRB)
W#16#4571	STOP, вызванный ошибкой скобочного стека
W#16#4572	STOP, вызванный ошибкой стека главного управляющего реле
W#16#4573	STOP из-за превышения глубины вложения при синхронных ошибках
W#16#4574	STOP, вызванный превышением глубины вложения стека прерываний в стеке классов приоритета
W#16#4575	STOP, вызванный превышением глубины вложения стека блоков в стеке классов приоритета

ID события	Событие
W#16#4576	STOP, вызванный ошибкой при размещении локальных данных
W#16#4578	STOP, вызванный неизвестным кодом операции
W#16#457A	STOP, вызванный ошибкой длины кода
W#16#457B	STOP, вызванный незагруженным DB у встроенной периферии
W#16#457F	STOP, вызванный командой STOP
W#16#4580	STOP: содержимое резервного буфера противоречиво (нет перехода в RUN)
W#16#4590	STOP, вызванный перегрузкой внутренних функций
W#16#49A0	STOP, вызванный ошибкой назначения параметров или недопустимым расхождением заданного и фактического расширения: запуск заблокирован.
W#16#49A1	STOP, вызванный ошибкой назначения параметров: запрос на сброс памяти
W#16#49A2	STOP, вызванный ошибкой при модификации параметров: запуск заблокирован
W#16#49A3	STOP, вызванный ошибкой при модификации параметров: запрос на сброс памяти
W#16#49A4	STOP: противоречивость данных конфигурирования
W#16#49A5	STOP: децентрализованная периферия: противоречия в загруженной проектной информации
W#16#49A6	STOP: децентрализованная периферия: недопустимая проектная информация
W#16#49A7	STOP: децентрализованная периферия: отсутствует проектная информация
W#16#49A8	STOP: ошибка, отображаемая интерфейсным модулем для децентрализованной периферии
W#16#43B0	Обновление программы ПЗУ было успешным
W#16#49B1	Ошибочные данные при обновлении программы ПЗУ
W#16#49B2	Обновление программы ПЗУ: версия аппаратуры не соответствует программе ПЗУ
W#16#49B3	Обновление программы ПЗУ: тип модуля не соответствует программе ПЗУ
W#16#49D0	LINK-UP (соединение) прервано из-за нарушения правил координации
W#16#49D1	LINK-UP/UPDATE (соединение/обновление) прервано
W#16#49D2	Резервный CPU перешел в STOP из-за перехода в STOP в основном CPU при соединении
W#16#43D3	STOP в резервном CPU, запрошенный пользователем
W#16#49D4	STOP в главном CPU, так как CPU-партнер тоже является главным (ошибка соединения)
W#16#45D5	LINK-UP/UPDATE (соединение/обновление) прервано из-за недопустимой конфигурации памяти CPU подустройства
W#16#45D6	LINK-UP (соединение/обновление) прервано из-за противоречий в системной программе подустройства
W#16#49D7	LINK-UP (соединение/обновление) прервано из-за противоречий в коде программы пользователя на платах памяти
W#16#45D8	DEFECTIVE: неисправность аппаратуры, обнаруженная благодаря другим ошибкам
W#16#45D9	STOP из-за ошибки модуля синхронизации (SYNC)
W#16#45DA	STOP из-за ошибки синхронизации между N CPU

<b>ID события</b>	<b>Событие</b>
W#16#45DC	ABORT: аварийное прекращение работы во время соединения посредством переключения
W#16#45DD	Соединение разорвано в связи с выполнением тестирования или других интерактивных функций
W#16#45DE	Модифицирование прервано из-за превышения контролируемого времени, во время n-ной попытки, предпринимается новая попытка модификации
W#16#43E0	Выход из режима самостоятельной работы после соединения
W#16#43E1	Выход из режима соединения поле обновления
W#16#43E2	Переход из режима обновления в режим резервирования
W#16#43E3	Основной CPU: переход из режима резервирования в режим самостоятельной работы
W#16#43E4	Резервный CPU: выход из режима резервирования после режима поиска ошибок
W#16#43E5	Резервный CPU: выход из режима поиска ошибок после соединения или состояния STOP
W#16#43E6	Соединение аварийно завершено на резервном CPU
W#16#43E7	Обновление аварийно завершено на резервном CPU
W#16#43E8	Резервный CPU: выход из соединения после запуска
W#16#43E9	Резервный CPU: выход из запуска после обновления
W#16#43F1	Переключение "резервный-основной"
W#16#43F2	Соединение несовместимых H-CPU заблокировано системной программой

## 32.6 Класс событий 5 – События этапа выполнения

ID события	Событие
W#16#530D	Новая информация о запуске в состоянии STOP
W#16#5311	Запуск несмотря на отсутствие сообщения о готовности от модуля(ей)
W#16#5961	Ошибка назначения параметров
W#16#5962	Ошибка назначения параметров, препятствующая запуску
W#16#5963	Ошибка назначения параметров с запросом на сброс памяти
W#16#5966	Ошибка назначения параметров при переключении режимов
W#16#5969	Ошибка назначения параметров с блокировкой запуска
W#16#5371	Децентрализованная периферия: конец синхронизации с ведущее устройством DP
W#16#5979/5879	Диагностическое сообщение от интерфейса DP: светодиод EXTf вкл/выкл
W#16#597C	Команда общего управления DP (DP Global Control) не выполнена или отвергнута
W#16#5380	Заблокированы записи в диагностический буфер прерываний и асинхронных ошибок
W#16#5395	Децентрализованная периферия: сброс ведущего DP-устройства
W#16#596	Ошибка назначения параметров при переключении режимов

## 32.7 Класс событий 6 – Коммуникационные события

ID события	Событие
W#16#6500	Идентификатор соединения содержится в модуле дважды
W#16#6501	Ресурсов связи недостаточно
W#16#6502	Ошибка в описании соединения
W#16#6905/6805	Проблема с ресурсами в сконфигурированных соединениях/устранена
W#16#6510	Ошибка структуры CFB в экземплярном DB при анализе СППЗУ
W#16#6514	Номер пакета GD содержится в модуле дважды
W#16#6515	Некорректное задание длины в проектной информации GD
W#16#6316	Ошибка интерфейса при запуске программируемого контроллера
W#16#6521	Ни submodule памяти, ни внутренняя память недоступны
W#16#6522	Недопустимый submodule памяти: замените submodule и сбросьте память
W#16#6523	Запрос на сброс памяти из-за ошибки при обращении к submodule
W#16#6524	Запрос на сброс памяти из-за ошибки в заголовке блока
W#16#6526	Запрос на сброс памяти из-за замены памяти
W#16#6527	Память заменена, поэтому рестарт невозможен
W#16#6528	Функция обработки объекта в режиме STOP/HOLD, рестарт невозможен
W#16#6529	Запуск невозможен при выполнении команды "Загрузить программу пользователя"
W#16#652A	Нет запуска, т.к. блок содержится дважды в памяти пользователя
W#16#652B	Нет запуска, т.к. длина блока слишком велика для submodule – замените submodule
W#16#652C	Нет запуска из-за недопустимого OB в submodule
W#16#6532	Нет запуска из-за недопустимой информации о конфигурации в submodule
W#16#6533	Запрос на сброс памяти из-за недопустимого содержимого submodule
W#16#6534	Нет запуска: блок содержится в submodule многократно
W#16#6535	Нет запуска: недостаточен объем памяти, чтобы передать блок из submodule
W#16#6536	Нет запуска: submodule содержит недопустимый номер блока
W#16#6537	Нет запуска: submodule содержит блок недопустимой длины
W#16#6538	Локальные данные или идентификатор защиты от записи (для DB) блока недопустимы для CPU
W#16#6539	В блоке недопустимая команда (выявлена компилятором)
W#16#653A	Запрос на сброс памяти, т.к. локальные данные OB в submodule слишком кратки
W#16#6543	Нет запуска: недопустимый тип блока
W#16#6544	Нет запуска: атрибут "существенно для обработки" недопустим
W#16#6545	Исходный язык недопустим
W#16#6546	Достигнуто максимальное количество проектной информации



<b>ID события</b>	<b>Событие</b>
W#16#6547	Ошибка при назначении параметров модулям (не на P-шине, отмените загрузку)
W#16#6548	Ошибка достоверности при проверке блока
W#16#6549	Структурная ошибка в блоке
W#16#6550	Блок имеет ошибку в CRC
W#16#6551	У блока нет CRC
W#16#6560	Переполнение SCAN
W#16#6981	Приходящая ошибка интерфейса
W#16#6881	Уходящая ошибка интерфейса
W#16#6390	Форматирование модуля памяти (Micro Memory Card) завершено

## 32.8 Класс событий 7 - События Н/Ф

ID события	Событие	ОВ
W#16#72A2	Отказ ведущего DP-устройства или ведущей DP-системы	ОВ70
W#16#72A3	Резервирование восстановлено на ведомом DP-устройстве	
W#16#7310/7210	Потеря резервирования системы ввода/вывода	
W#16#7311/7211	Частичная потеря резервирования системы ввода/вывода	
W#16#73A3	Потеря резервирования на ведомом DP-устройстве	
W#16#7301	Потеря резервирования (1 из 2) вследствие отказа CPU	ОВ72
W#16#7302	Потеря резервирования (1 из 2) из-за запущенного пользователем перехода в STOP в резерве	
W#16#7303	Н-система (1 из 2) перешла в режим резервирования	
W#16#7520	Ошибка при сравнении RAM	
W#16#7521	Ошибка при сравнении значения выходов образа процесса	
W#16#7522	Ошибка при сравнении битов памяти, таймеров или счетчиков	
W#16#7323	Обнаружено противоречие в данных операционной системы	
W#16#7331	Переключение на резервное ведущее устройство из-за отказа master-устройства	
W#16#7333	Переключение на резервное ведущее устройство из-за вмешательства оператора	
W#16#7934	Переключение на резервное ведущее устройство из-за проблемы подключения в модуле синхронизации	
W#16#7335	Переключение на резервное ведущее устройство, запущенное SFC90 "H_CTRL"	
W#16#7340	Ошибка синхронизации в программе пользователя из-за истекшего времени ожидания	
W#16#7341	Ошибка синхронизации в программе пользователя из-за ожидания в различных точках синхронизации	
W#16#7342	Ошибка синхронизации в операционной системе из-за ожидания в различных точках синхронизации	
W#16#7343	Ошибка синхронизации в операционной системе из-за истекшего времени ожидания	
W#16#7344	Ошибка синхронизации в операционной системе из-за неправильных данных	
W#16#734A	Было выполнено задание на "депассивацию", запущенное SFC90 "H_CTRL"	
W#16#73C1	Процесс обновления прерван	
W#16#73C2	Обновление завершен из-за превышения времени мониторинга во время n-ной попытки (1 < n < максимально возможного числа попыток выполнить обновление)	
W#16#7950	Модуль синхронизации не существует	
W#16#7951	Изменение в модуле синхронизации без включения питания	
W#16#7952/7852	Модуль синхронизации удален/вставлен	

<b>ID события</b>	<b>Событие</b>	<b>ОВ</b>
W#16#7953	Изменение в модуле синхронизации без сброса	OB72
W#16#7954	Модуль синхронизации: номер стойки назначен дважды	
W#16#7955/7855	Ошибка модуля синхронизации/устранена	
W#16#7956	В модуле синхронизации установлен запрещенный номер стойки	
W#16#73E0/72E0	Потеря резервирования в системе связи / резервирование восстановлено	OB73
W#16#734A	Было выполнено задание на "депассивацию", запущенное SFC90 "H_CTRL"	
W#16#734B	Было выполнено задание на "депассивацию", запущенное операционной системой	

## 32.9 Класс событий 8 – Диагностические события для модулей

ID события	Событие	Тип модуля	
W#16#8x00	Модуль неисправен/ исправен	Любой	
W#16#8x01	Внутренняя ошибка		
W#16#8x02	Внешняя ошибка		
W#16#8x03	Ошибка канала		
W#16#8x04	Отсутствует внешнее вспомогательное напряжение		
W#16#8x05	Отсутствует фронт-штекер		
W#16#8x06	Параметры не назначены		
W#16#8x07	Неверные параметры в модуле		
W#16#8x30	Субмодуль пользователя неисправен/ не найден		
W#16#8x31	Неисправность связи		
W#16#8x32	Режим работы: RUN/STOP (STOP: наступающий, RUN: уходящий)		
W#16#8x33	Сработал контроль времени (сторожевая схема)		
W#16#8x34	Вышло из строя внутренний источник питания модуля		
W#16#8x35	BATTF: батарея разряжена		
W#16#8x36	Вышла из строя вся буферизация		
W#16#8x37	Резерв		
W#16#8x40	Вышла из строя стойка расширения		
W#16#8x41	Вышел из строя процессор		
W#16#8x42	Ошибка СППЗУ		
W#16#8x43	Ошибка ОЗУ		
W#16#8x44	Ошибка АЦП/ЦАП		
W#16#8x45	Сгорел предохранитель		
W#16#8x46	Потеряно аппаратное прерывание		
W#16#8x47	Резерв		
W#16#8x50	Ошибка конфигурирования/ параметризации		Аналоговый вход
W#16#8x51	Синфазная ошибка		
W#16#8x52	Короткое замыкание на фазу		
W#16#8x53	Короткое замыкание на землю		
W#16#8x54	Обрыв провода		
W#16#8x55	Ошибка опорного канала		
W#16#8x56	Переход нижней границы измерения		
W#16#8x57	Переход верхней границы измерения		

ID события	Событие	Тип модуля
W#16#8x60	Ошибка конфигурирования/ параметризации	Аналоговый выход
W#16#8x61	Синфазная ошибка	
W#16#8x62	Короткое замыкание на фазу	
W#16#8x63	Короткое замыкание на землю	
W#16#8x64	Обрыв провода	
W#16#8x65	Резерв	
W#16#8x66	Отсутствует напряжение на зажимах	Цифровой ввод
W#16#8x70	Ошибка конфигурирования/ параметризации	
W#16#8x71	Неисправность заземления на корпус	
W#16#8x72	Короткое замыкание на фазу (датчик)	
W#16#8x73	Короткое замыкание на землю (датчик)	
W#16#8x74	Обрыв провода	
W#16#8x75	Отсутствует питание датчика	Цифровой вывод
W#16#8x80	Ошибка конфигурирования/ параметризации	
W#16#8x81	Неисправность заземления на корпус	
W#16#8x82	Короткое замыкание на фазу	
W#16#8x83	Короткое замыкание на землю	
W#16#8x84	Обрыв провода	
W#16#8x85	Обрыв цепи плавкого предохранителя	FM
W#16#8x86	Нет напряжения на зажимах	
W#16#8x87	Превышение температуры	
W#16#8xB0	Счетчик, ошибка в сигнале А	
W#16#8xB1	Счетчик, ошибка в сигнале В	
W#16#8xB2	Счетчик, ошибка в сигнале N	
W#16#8xB3	Счетчик, между каналами передано ошибочное значение	
W#16#8xB4	Счетчик, неисправно питание датчика 5,2 В	
W#16#8xB5	Счетчик, неисправно питание датчика 24 В	

### 32.10 Класс событий 9 – Стандартные события пользователя

ID события	Событие
W#16#9001	Автоматический режим
W#16#9101	Ручной режим
W#16#9x02	ОТКРЫТО/ЗАКРЫТО, ВКЛ/ВЫКЛ
W#16#9x03	Разрешение ручного управления
W#16#9x04	Команда защиты агрегата (ОТКРЫТО/ЗАКРЫТО)
W#16#9x05	Деблокировка процесса
W#16#9x06	Команда защиты системы
W#16#9x07	Сработал контроль переменной процесса
W#16#9x08	Сработал контроль заданного значения
W#16#9x09	Ошибка регулирования больше допустимой
W#16#9x0A	Ошибка граничного состояния
W#16#9x0B	Динамическая ошибка
W#16#9x0C	Ошибка исполнения команды (генератор последовательности)
W#16#9x0D	Рабочий режим выполняется > ОТКРЫТО
W#16#9x0E	Рабочий режим выполняется > ЗАКРЫТО
W#16#9x0F	Блокирование команды
W#16#9x11	Состояние процесса ОТКРЫТО /ВКЛЮЧЕНО
W#16#9x12	Состояние процесса ЗАКРЫТО / ВЫКЛЮЧЕНО
W#16#9x13	Состояние процесса промежуточное
W#16#9x14	Состояние процесса ВКЛЮЧЕНО через АВТОМАТИЧЕСКИЙ РЕЖИМ
W#16#9x15	Состояние процесса ВКЛЮЧЕНО через РУЧНОЙ РЕЖИМ
W#16#9x16	Состояние процесса ВКЛЮЧЕНО через команду защиты
W#16#9x17	Состояние процесса ВЫКЛЮЧЕНО через АВТОМАТИЧЕСКИЙ РЕЖИМ
W#16#9x18	Состояние процесса ВЫКЛЮЧЕНО через РУЧНОЙ РЕЖИМ
W#16#9x19	Состояние процесса ВЫКЛЮЧЕНО через команду защиты
W#16#9x21	Ошибка функционирования при приближении
W#16#9x22	Ошибка функционирования при удалении
W#16#9x31	Исполнительный орган (DE/WE) граничное положение ОТКРЫТО
W#16#9x32	Исполнительный орган (DE/WE) граничное положение не ОТКРЫТО
W#16#9x33	Исполнительный орган (DE/WE) граничное положение ЗАКРЫТО
W#16#9x34	Исполнительный орган (DE/WE) граничное положение не ЗАКРЫТО
W#16#9x41	Неразрешенное состояние, допустимое время истекло
W#16#9x42	Неразрешенное состояние, допустимое время не истекло
W#16#9x43	Ошибка блокировки, допустимое время = 0
W#16#9x44	Ошибка блокировки, допустимое время > 0

ID события	Событие
W#16#9x45	Реакция отсутствует
W#16#9x46	Неразрешенный выход из конечного состояния, допустимое время = 0
W#16#9x47	Неразрешенный выход из конечного состояния, допустимое время > 0
W#16#9x50	Верхняя граница диапазона сигнала USR
W#16#9x51	Верхняя граница диапазона измерения UMR
W#16#9x52	Нижняя граница диапазона сигнала LSR
W#16#9x53	Нижняя граница диапазона измерения LMR
W#16#9x54	Верхняя граница аварийного сигнала UAL
W#16#9x55	Верхняя граница предупреждения UWL
W#16#9x56	Верхняя допустимая граница UTL
W#16#9x57	Нижняя допустимая граница LTL
W#16#9x58	Нижняя граница предупреждения LWL
W#16#9x59	Нижняя граница аварийного сигнала LAL
W#16#9x60	Шаг GRAPH7 наступающий/уходящий
W#16#9x61	Ошибка блокировки GRAPH7
W#16#9x62	Ошибка исполнения GRAPH7
W#16#9x63	Ошибка GRAPH7 принята к сведению
W#16#9x64	Ошибка GRAPH7 квитируется
W#16#9x70	Среднее значение перейдено в положительном направлении
W#16#9x71	Среднее значение перейдено в отрицательном направлении
W#16#9x72	Реакция отсутствует
W#16#9x73	Недопустимый выход из конечного состояния
W#16#9x80	Выход за верхнюю границу, допустимое время = 0
W#16#9x81	Выход за верхнюю границу, допустимое время > 0
W#16#9x82	Выход за нижнюю границу, допустимое время = 0
W#16#9x83	Выход за нижнюю границу, допустимое время > 0
W#16#9x84	Выход за верхнюю границу градиента, допустимое время = 0
W#16#9x85	Выход за верхнюю границу градиента, допустимое время > 0
W#16#9x86	Выход за нижнюю границу градиента, допустимое время = 0
W#16#9x87	Выход за нижнюю границу градиента, допустимое время > 0
W#16#9190/9090	Ошибка назначения параметров пользователя наступающая/ уходящая
W#16#91F0	Переполнение
W#16#91F1	Потеря значимости
W#16#91F2	Деление на ноль
W#16#91F3	Недопустимая математическая операция

### 32.11 Классы событий А и В – Свободные пользовательские события

ID события	Событие
W#16#Axyz	События, доступные для пользователя
W#16#Bxyz	



## 32.12 Резервные классы событий

### Зарезервированные классы событий

Следующие классы событий зарезервированы для последующих расширений:

- C
- D
- E
- F Зарезервировано для модулей, не находящихся в центральной стойке (например, для модулей CP или FM)



## **33 Списки SFC и SFB**

### 33.1 Список SFC, упорядоченный по номерам

№	Краткое название	Функция
SFC0	SET_CLK	Установка системных часов
SFC1	READ_CLK	Чтение системных часов
SFC2	SET_RTM	Установка счетчика рабочего времени
SFC3	CTRL_RTM	Запуск/останов счетчика рабочего времени
SFC4	READ_RTM	Чтение счетчика рабочего времени
SFC5	GADR_LGC	Выяснение логического адреса канала
SFC6	RD_SINFO	Считывание стартовой информации ОВ
SFC7	DP_PRAL	Запуск аппаратного прерывания на ведущем DP-устройстве
SFC9	EN_MSG	Разрешение сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии
SFC10	DIS_MSG	Блокирование сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии
SFC11	DPSYC_FR	Синхронизация групп ведомых DP-устройств
SFC12	D_ACT_DP	Деактивация и активация ведомых DP-устройств
SFC13	DPNRM_DG	Чтение диагностических данных ведомые DP-устройства (Slave-диагностика)
SFC14	DPRD_DAT	Чтение консистентных данных стандартного ведомые DP-устройства
SFC15	DPWR_DAT	Запись консистентных данных в стандартное ведомое DP-устройство
SFC17	ALARM_SQ	Генерирование квитуемых сообщений, связанных с блоками
SFC18	ALARM_S	Генерирование постоянно квитуемых сообщений, связанных с блоками
SFC19	ALARM_SC	Определение состояния квитирования последнего пришедшего сообщения ALARM_SQ
SFC20	BLKMOV	Копирование переменных
SFC21	FILL	Инициализация области памяти
SFC22	CREAT_DB	Создание блока данных
SFC23	DEL_DB	Удаление блока данных
SFC24	TEST_DB	Тестирование блока данных
SFC25	COMPRESS	Сжатие памяти пользователя
SFC26	UPDAT_PI	Обновление таблицы входов образа процесса
SFC27	UPDAT_PO	Обновление таблицы выходов образа процесса
SFC28	SET_TINT	Установка прерывания по времени
SFC29	CAN_TINT	Отмена прерывания по времени
SFC30	ACT_TINT	Активация прерывания по времени
SFC31	QRY_TINT	Опрос прерывания по времени
SFC32	SRT_DINT	Запуск прерывания с задержкой
SFC33	CAN_DINT	Отмена прерывания с задержкой

№	Краткое название	Функция
SFC34	QRY_DINT	Опрос прерывания с задержкой
SFC35	MP_ALM	Запуск мультипроцессорного прерывания
SFC36	MSK_FLT	Маскирование синхронных ошибок
SFC37	DMSK_FLT	Демаскирование синхронных ошибок
SFC38	READ_ERR	Чтение регистра ошибок
SFC39	DIS_IRT	Блокирование новых прерываний и асинхронных ошибок
SFC40	EN_IRT	Деблокирование новых прерываний и асинхронных ошибок
SFC41	DIS_AIRT	Задержка более приоритетных прерываний и асинхронных ошибок
SFC42	EN_AIRT	Деблокирование более приоритетных прерываний и асинхронных ошибок
SFC43	RE_TRIGR	Повторный запуск контроля времени цикла
SFC44	REPL_VAL	Передача заменяющего значения в аккумулятор 1
SFC46	STP	Перевод CPU в состояние STOP
SFC47	WAIT	Задержка исполнения программы пользователя
SFC48	SNC_RTCB	Синхронизация ведомых часов
SFC49	LGC_GADR	Выявление слота модуля, соответствующего логическому адресу
SFC50	RD_LGADR	Выявление всех логических адресов модуля
SFC51	RDSYSST	Считывание списка и подписка состояний системы
SFC52	WR_USMSG	Запись в диагностический буфер диагностического события, определенного пользователем
SFC54	RD_PARM	Считывание определенных параметров
SFC55	WR_PARM	Запись динамических параметров
SFC56	WR_DPARM	Запись параметров по умолчанию
SFC57	PARAM_MOD	Назначение параметров модулю
SFC58	WR_REC	Внесение записи данных
SFC59	RD_REC	Чтение записи данных
SFC60	GD_SND	Передача GD-пакета
SFC61	GD_RCV	Извлечение принятого GD-пакета
SFC62	CONTROL	Опрос состояния соединения, принадлежащего экземпляру коммуникационного SFB
SFC63	AB_CALL	Вызов блока скомпонованного кода
SFC64	TIME_TCK	Чтение системного времени
SFC65	X_SEND	Передача данных партнеру по связи вне локальной станции S7
SFC66	X_RCV	Прием данных от партнера по связи вне локальной станции S7
SFC67	X_GET	Чтение данных от партнера по связи вне локальной станции S7
SFC68	X_PUT	Запись данных в партнера по связи вне локальной станции S7
SFC69	X_ABORT	Прерывание существующего соединения с партнером по связи вне локальной станции S7

№	Краткое название	Функция
SFC72	I_GET	Чтение данных от партнера по связи внутри локальной станции S7
SFC73	I_PUT	Запись данных партнеру по связи внутри локальной станции S7
SFC74	I_ABORT	Прерывание существующего соединения с партнером по связи внутри локальной станции S7
SFC78	OB_RT	Определение времени выполнения программы OB
SFC79	SET	Установка области выходов
SFC80	RSET	Сброс области выходов
SFC81	UBLKMOV	Непрерываемое копирование переменных
SFC82	CREA_DBL	Генерация блока данных в загружаемой памяти
SFC83	READ_DBL	Чтение из блока данных в загружаемой памяти
SFC84	WRIT_DBL	Запись в блоке данных в загружаемой памяти
SFC85	CREA_DB	Создание блока данных
SFC87	C_DIAG	Диагностика фактического состояния соединения
SFC90	H_CTRL	Управление функционированием в H-системах
SFC100	SET_CLKS	Установка времени суток и состояния таймера (TOD Status)
SFC101	RTM	Управление счетчиками времени выполнения (runtime)
SFC102	RD_DPARA	Переопределенные параметры
SFC103	DB_TOPOL	Идентификация шинной топологии с системе ведущего DP-устройства
SFC104	CiR	Управление CiR
SFC105	READ_SI	Считывание динамически занимаемых системных ресурсов
SFC106	DEL_SI	Удаление динамически занимаемых системных ресурсов
SFC107	ALARM_DQ	Генерация квитуемых сообщений, связанных с блоком
SFC108	ALARM_D	Генерация всегда квитуемых сообщений, связанных с блоком
SFC112	PN_IN	Обновление входов в UPI для PROFINet-компонентов
SFC113	PN_OUT	Обновление выходов в UPI для PROFINet-компонентов
SFC114	PN_DP	Обновление взаимных соединений в системе распределенной периферии DP
SFC126	SYNC_PI	Обновление таблицы входов раздела отображения процесса в синхронном цикле
SFC127	SYNC_PO	Обновление таблицы выходов раздела отображения процесса в синхронном цикле

\* SFC63 "AB\_CALL" существует только для CPU 614. За подробным описанием обратитесь к соответствующему руководству.

### 33.2 Список SFC, упорядоченный по алфавиту

Краткое название	№	Функция
AB_CALL	SFC63	Вызов блока скомпонованного кода
ACT_TINT	SFC30	Активация прерывания по времени
ALARM_D	SFC 108	Генерация всегда квитируемых сообщений, связанных с блоком
ALARM_DQ	SFC 107	Генерация квитируемых сообщений, связанных с блоком
ALARM_SQ	SFC17	Генерирование квитируемых сообщений, связанных с блоками
ALARM_S	SFC18	Генерирование постоянно квитируемых сообщений, связанных с блоками
ALARM_SC	SFC19	Определение состояния квитирувания последнего пришедшего сообщения ALARM_SQ
BLKMOV	SFC20	Копирование переменных
C_DIAG	SFC 87	Диагностика фактического состояния соединения
CAN_DINT	SFC33	Отмена прерывания с задержкой
CAN_TINT	SFC29	Отмена прерывания по времени
COMPRESS	SFC25	Сжатие памяти пользователя
CiR	SFC104	Управление CiR
CONTROL	SFC62	Опрос состояния соединения, принадлежащего экземпляру коммуникационного SFB
CREA_DB	SFC85	Создание блока данных
CREA_DBL	SFC 82	Генерация блока данных в загружаемой памяти
CREAT_DB	SFC22	Создание блока данных
CTRL_RTM	SFC3	Запуск/останов счетчика рабочего времени
D_ACT_DP	SFC 12	Деактивация/активация ведомых DP-устройств
DEL_DB	SFC23	Удаление блока данных
DEL_SI	SFC106	Очистка динамически занимаемых системных ресурсов
DIS_AIRT	SFC41	Задержка более приоритетных прерываний и асинхронных ошибок
DIS_IRT	SFC39	Блокирование новых прерываний и асинхронных ошибок
DIS_MSG	SFC10	Блокирование сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии
DMSK_FLT	SFC37	Демаширование синхронных ошибок
DP_PRAL	SFC7	Запуск аппаратного прерывания на ведущем DP-устройстве
DB_TOPOL	SFC103	Идентификация шинной топологии с системе ведущего DP-устройства
DPNRM_DG	SFC13	Чтение диагностических данных ведомые DP-устройства (Slave-диагностика)
DPRD_DAT	SFC14	Чтение консистентных данных стандартного ведомые DP-устройства
DPSYC_FR	SFC11	Синхронизация групп ведомых DP-устройств
DPWR_DAT	SFC15	Запись консистентных данных в стандартное ведомое DP-устройство
EN_AIRT	SFC42	Деблокирование более приоритетных прерываний и асинхронных ошибок

Краткое название	№	Функция
EN_IRT	SFC40	Деблокирование новых прерываний и асинхронных ошибок
EN_MSG	SFC9	Деблокирование сообщений, связанных с блоком, сообщений, связанных с символом, и сообщений о групповом состоянии
FILL	SFC21	Инициализация области памяти
GADR_LGC	SFC5	Выяснение логического адреса канала
GD_RCV	SFC61	Извлечение принятого GD-пакета
GD_SND	SFC60	Передача GD-пакета
H_CTRL	SFC90	Управление функционированием в H-системах
I_ABORT	SFC74	Прерывание существующей связи с коммуникационным партнером внутри локальной станции S7
I_GET	SFC72	Чтение данных из коммуникационного партнера внутри локальной станции S7
I_PUT	SFC73	Запись данных в коммуникационного партнера внутри локальной станции S7
LGC_GADR	SFC49	Запрос слота модуля по логическому адресу
MP_ALM	SFC35	Запуск прерывания мультипроцессорной обработки
MSK_FLT	SFC36	Маскирование синхронных ошибок
PARM_MOD	SFC57	Назначение параметров модулю
PN_DP	SFC114	Обновление взаимных соединений в системе распределенной периферии DP
PN_IN	SFC112	Обновление входов в UPI для PROFINet-компонентов
PN_OUT	SFC113	Обновление выходов в UPI для PROFINet-компонентов
QRY_DINT	SFC34	Опрос прерывания с задержкой
QRY_TINT	SFC31	Опрос прерывания по времени
RD_DPARA	SFC102	Переопределение параметров
RD_PARM	SFC54	Считывание определенных параметров
RD_LGADR	SFC50	Выявление всех логических адресов модуля
RD_REC	SFC59	Чтение записи данных
RD_SINFO	SFC6	Считывание стартовой информации OB
RDSYSST	SFC51	Считывание списка и подписки состояний системы
RE_TRIGR	SFC43	Повторный запуск контроля времени цикла
READ_CLK	SFC1	Чтение системных часов
READ_DBL	SFC83	Чтение из блока данных в загружаемой памяти
READ_RTM	SFC4	Чтение счетчика рабочего времени
READ_ERR	SFC38	Чтение регистра ошибок
READ_SI	SFC105	Чтение динамически занимаемых ресурсов
REPL_VAL	SFC44	Передача заменяющего значения в аккумулятор 1
RSET	SFC80	Сброс области выходов
RTM	SFC101	Управление счетчиками времени выполнения (runtime)
SET	SFC79	Установка области выходов
SET_CLK	SFC0	Установка системных часов
SET_CLKS	SFC100	Установка времени суток и состояния таймера (TOD Status)

Системные и стандартные функции для S7-300 и S7-400  
A5E00261410-01



Краткое название	№	Функция
SET_RTM	SFC2	Установка счетчика рабочего времени
SET_TINT	SFC28	Установка прерывания по времени
SNC_RTCB	SFC48	Синхронизация ведомых часов
SRT_DINT	SFC32	Запуск прерывания с задержкой
STP	SFC46	Перевод CPU в состояние STOP
SYNC_PI	SFC 126	Обновление таблицы входов раздела отображения процесса в синхронном цикле
SYNC_PO	SFC 127	Обновление таблицы выходов раздела отображения процесса в синхронном цикле
TEST_DB	SFC24	Тестирование блока данных
TIME_TCK	SFC64	Чтение системного времени
UBKLMOV	SFC81	Непрерываемое копирование переменных
UPDAT_PI	SFC26	Обновление таблицы входов образа процесса
UPDAT_PO	SFC27	Обновление таблицы выходов образа процесса
WAIT	SFC47	Задержка исполнения программы пользователя
WR_DPARM	SFC56	Запись параметров по умолчанию
WR_PARM	SFC55	Запись динамических параметров
WR_REC	SFC58	Внесение записи данных
WR_USMSG	SFC52	Запись в диагностический буфер диагностического события, определенного пользователем
WRIT_DBL	SFC84	Запись в блоке данных в загружаемой памяти
X_ABORT	SFC69	Прерывание существующего соединения с партнером по связи вне локальной станции S7
X_GET	SFC67	Чтение данных от партнера по связи вне локальной станции S7
X_PUT	SFC68	Запись данных в партнера по связи вне локальной станции S7
X_RCV	SFC66	Прием данных от партнера по связи вне локальной станции S7
X_SEND	SFC65	Передача данных партнеру по связи вне локальной станции S7

\* SFC63 "AB\_CALL" существует только для CPU 614. За подробным описанием обратитесь к соответствующему руководству.

### 33.3 Список SFB, упорядоченный по номерам

№	Краткое название	Функция
SFB0	CTU	Прямой счет
SFB1	CTD	Обратный счет
SFB2	CTUD	Прямой и обратный счет
SFB3	TP	Генерирование импульса
SFB4	TON	Генерирование задержки включения
SFB5	TOF	Генерирование задержки выключения
SFB8	USEND	Некоординированная передача данных
SFB9	URCV	Некоординированный прием данных
SFB12	BSEND	Передача сегментированных данных
SFB13	BRCV	Прием сегментированных данных
SFB14	GET	Чтение данных из удаленного CPU
SFB15	PUT	Запись данных в удаленный CPU
SFB16	PRINT	Передача данных на принтер
SFB19	START	Инициализация полного рестарта в удаленном устройстве
SFB20	STOP	Перевод удаленного устройства в состояние STOP
SFB21	RESUME	Инициализация рестарта в удаленном устройстве
SFB22	STATUS	Опрос состояния удаленного партнера
SFB23	USTATUS	Получение состояния удаленного устройства
SFB29	HS_COUNT	Счетчик (высокоскоростной счетчик, встроенная функция)
SFB30	FREQ_MES	Частотомер (измеритель частоты, встроенная функция)
SFB31	NOTIFY_8P	Генерирование сообщений, связанных с блоком, без индикации квитирования
SFB32	DRUM	Реализация генератора последовательностей
SFB33	ALARM	Генерирование сообщений, связанных с блоком, с индикацией квитирования
SFB34	ALARM_8	Генерирование сообщений, связанных с блоком, без сопутствующих значений для восьми сигналов
SFB35	ALARM_8P	Генерирование сообщений, связанных с блоком, с сопутствующими значениями для восьми сигналов
SFB36	NOTIFY	Генерирование сообщений, связанных с блоком, без индикации квитирования
SFB37	AR_SEND	Передача архивных данных
SFB38	HSC_A_B	Счетчик A/B (встроенная функция)
SFB39	POS	Позиционирование (встроенная функция)
SFB41	CONT_C <sup>1)</sup>	Непрерывное регулирование

№	Краткое название	Функция
SFB42	CONT_S <sup>1)</sup>	Ступенчатое регулирование
SFB43	PULSEGEN <sup>1)</sup>	Генерирование импульсов
SFB44	ANALOG <sup>2)</sup>	Позиционирование при аналоговом выходе
SFB46	DIGITAL <sup>2)</sup>	Позиционирование при дискретном выходе
SFB47	COUNT <sup>2)</sup>	Управление счетчиком
SFB48	FREQUENC <sup>2)</sup>	Управление измерением частоты
SFB49	PULSE <sup>2)</sup>	Управление ШИМ-модуляцией
SFB52	RDREC	Считывание записи данных из ведомого DP-устройства
SFB53	WRREC	Внесение записи данных в ведомое DP-устройство
SFB54	RALRM	Прием прерывания от ведомого DP-устройства
SFB60	SEND_PTP <sup>2)</sup>	Посылка данных (ASCII, 3964(R))
SFB61	RECV_PTP <sup>2)</sup>	Прием данных (ASCII, 3964(R))
SFB62	RES_RECV <sup>2)</sup>	Очистка входного буфера (ASCII, 3964(R))
SFB63	SEND_RK <sup>2)</sup>	Посылка данных (RK 512)
SFB64	FETCH_RK <sup>2)</sup>	Выборка данных (RK 512)
SFB65	SERVE_RK <sup>2)</sup>	Прием и выдача данных (RK 512)
SFB75	SALRM	Посылка прерывания ведущему DP-устройству

\* SFB29 "HS\_COUNT" и SFB30 "FREQ\_MES" имеются только в CPU 312 IFM и CPU 314 IFM. SFB 38 "HSC\_A\_B" и 39 "POS" имеются только в CPU 314 IFM. За подробным описанием обратитесь к [I73!](#)

<sup>1)</sup> SFB 41 "CONT\_C," 42 "CONT\_S" и 43 "PULSEGEN" имеются только в CPU 314 IFM.

<sup>2)</sup> SFB 44 ... 49 и 60 ... 65 имеются только в CPU для S7-300C.

### 33.4 Список SFB, упорядоченный по алфавиту

Краткое название	№	Функция
ALARM	SFB33	Генерирование сообщений, связанных с блоком, с индикацией квитирования
ALARM_8	SFB34	Генерирование сообщений, связанных с блоком, без сопутствующих значений для восьми сигналов
ALARM_8P	SFB35	Генерирование сообщений, связанных с блоком, с сопутствующими значениями для восьми сигналов
ANALOG	SFB44	Позиционирование при аналоговом выходе
AR_SEND	SFB37	Передача архивных данных
BRCV	SFB13	Прием сегментированных данных
BSEND	SFB12	Передача сегментированных данных
CONT_C <sup>1)</sup>	SFB41	Непрерывное регулирование
CONT_S <sup>1)</sup>	SFB42	Ступенчатое регулирование
COUNT	SFB47	Управление счетчиком
CTD	SFB1	Обратный счет
CTU	SFB0	Прямой счет
CTUD	SFB2	Прямой и обратный счет
DIGITAL	SFB46	Позиционирование при дискретном выходе
DRUM	SFB32	Реализация генератора последовательностей
FETCH_RK	SFB64	Выборка данных (RK 512)
FREQ_MES	SFB30	Частотомер (измеритель частоты, встроенная функция)
FREQUENC	SFB48	Управление измерением частоты
GET	SFB14	Чтение данных из удаленного CPU
HSC_A_B	SFB38	Счетчик A/B (встроенная функция)
HS_COUNT	SFB29	Счетчик (высокоскоростной счетчик, встроенная функция)
NOTIFY	SFB36	Генерирование сообщений, связанных с блоком, без индикации квитирования
NOTIFY_8P	SFB31	Генерирование сообщений, связанных с блоком, без индикации квитирования
POS	SFB39	Позиционирование (встроенная функция)
PRINT	SFB16	Передача данных на принтер
PULSE	SFB49	Управление ШИМ-модуляцией
PULSEGEN <sup>1)</sup>	SFB43	Генерирование импульсов
PUT	SFB15	Запись данных в удаленный CPU
RALRM	SFB54	Прием прерывания от ведомого DP-устройства
RDREC	SFB52	Считывание записи данных их ведомого DP-устройства

Краткое название	№	Функция
RECV_PTP	SFB61	Прием данных (ASCII, 3964(R))
RES_RECV	SFB62	Очистка входного буфера (ASCII, 3964(R))
RESUME	SFB21	Инициализация рестарта в удаленном устройстве
SALRM	SFB75	Посылка прерывания ведущему DP-устройству
SEND_PTP	SFB60	Посылка данных (ASCII, 3964(R))
SEND_RK	SFB63	Посылка данных (RK 512)
SERVE_RK	SFB65	Прием и выдача данных (RK 512)
START	SFB19	Инициализация полного рестарта в удаленном устройстве
STATUS	SFB22	Опрос состояния удаленного партнера
STOP	SFB20	Перевод удаленного устройства в состояние STOP
TOF	SFB5	Генерирование задержки выключения
TON	SFB4	Генерирование задержки включения
TP	SFB3	Генерирование импульса
URCV	SFB9	Некоординированный прием данных
USEND	SFB8	Некоординированная передача данных
USTATUS	SFB23	Получение состояния удаленного устройства

\* SFB29 "HS\_COUNT" и SFB30 "FREQ\_MES" имеются только в CPU 312 IFM и CPU 314 IFM. SFB 38 "HSC\_A\_B" и 39 "POS" имеются только в CPU 314 IFM. За подробным описанием обратитесь к [/73/](#).

1) SFB 41 "CONT\_C," 42 "CONT\_S" и 43 "PULSEGEN" имеются только в CPU 314 IFM.



# Глоссарий

## А

### Адрес

Адрес - это идентификатор, присваиваемый ячейке памяти или некоторой области ячеек памяти, например: вход I 12.1; меркерное слово MW25; блок данных DB3.

### Адресация

Назначение адреса в программе пользователя. Адреса могут быть назначены ячейкам памяти или областям ячеек памяти (например: вход I 12.1; меркерное слово MW25).

### Аккумулятор

Аккумуляторы - это регистры в CPU, которые служат в качестве промежуточной памяти при операциях загрузки, передачи, а также сравнения, преобразования и в арифметических операциях.

### Аппаратное прерывание

Аппаратное прерывание вызывается модулями, выполняющими прерывание на основе определенных событий в процессе. Аппаратное прерывание передается в CPU. Затем, в соответствии с приоритетом этого прерывания, обрабатывается соответствующий → **организационный блок**.

## Б

### Битовая память (меркер)

Битовая память - это память, допускающая побитовый доступ. Основные операции STEP 7 имеют доступ к битовой памяти на запись и чтение (с адресацией побитно, побайтно, пословно и двойными словами). Область битовой памяти может использоваться для сохранения промежуточных результатов.

**Блок данных (DB)**

Блоки данных - это области в программе пользователя, содержащие данные пользователя. Имеются совместно используемые (глобальные) блоки данных, к которым можно обратиться из любого кодового блока, и экземплярные блоки данных, которые ставятся в соответствие отдельному вызову функционального блока (FB).

**В****Встроенный регулятор**

Встроенный регулятор - это готовый запрограммированный блок регулирования, хранящийся в операционной системе и содержащий важнейшие функции автоматического управления. Пользователь с помощью программного переключателя может включать или выключать эти функции.

**Вспомогательный номер**

Номер контролируемого сигнала, если блок, формирующий сообщение, может вести мониторинг более, чем одного сигнала.

**Входные параметры**

Входные параметры имеются только у функций и функциональных блоков. С помощью входных параметров данные передаются для обработки в вызываемый блок.

**Г****Групповая ошибка**

Сообщение об ошибке, отображаемое светодиодами на передней панели модулей только в S7-300. Светодиод загорается при любой ошибке (внутренней или внешней) в соответствующем модуле.

**Групповые сообщения системы управления**

Групповые сообщения генерируются операционной системой CPU при появлении в диагностическом буфере информации о наступлении диагностического события.



## **Д**

### **Двухпозиционный регулятор**

Двухпозиционным регулятором называется регулятор, в котором управляющее воздействие может принимать только два состояния (напр., включено – выключено).

### **Диагностика**

Диагностические функции охватывают всю системную диагностику и включают в себя распознавание, интерпретацию и сообщение об ошибках внутри системы автоматизации.

### **Диагностическая запись**

Диагностическое событие описывается в диагностическом буфере с помощью диагностической записи.

### **Диагностический буфер**

Диагностический буфер - это область памяти в CPU, в которой хранятся все диагностические события в порядке их возникновения.

### **Диагностические данные**

Диагностические данные - это информация, содержащаяся в сообщении об ошибке (диагностическое событие, метка времени).

### **Диагностическое прерывание**

Модули, обладающие диагностическими свойствами, сообщают в → CPU о распознанных системных ошибках с помощью диагностических прерываний.

### **Диагностическое сообщение**

Диагностическое сообщение состоит из обработанного диагностического события и посылается из CPU на устройство отображения.

## **И**

### **Интегрирующее звено**

Интегрирующее звено регулятора. При скачкообразном изменении регулируемой величины (или сигнала ошибки) выходная величина растет линейно с течением времени со скоростью, пропорциональной  $K_I (=1/T_I)$ . Интегрирующее звено в замкнутом контуре регулирования действует так, что выходное значение регулируемой величины изменяется до тех пор, пока сигнал ошибки не станет равным нулю.

## Информация о стартовом событии

Информация о стартовом событии - это составная часть  
 → **организационного блока (ОВ)**. Она предоставляет пользователю S7 подробную информацию о событии, вызвавшем запуск этого ОВ. Информация о стартовом событии содержит номер события (состоящий из класса и идентификатора события), метку времени события, а также дополнительные данные (например, адрес сигнального модуля, вызвавшего прерывание).

## Исполнение программы, управляемое событиями

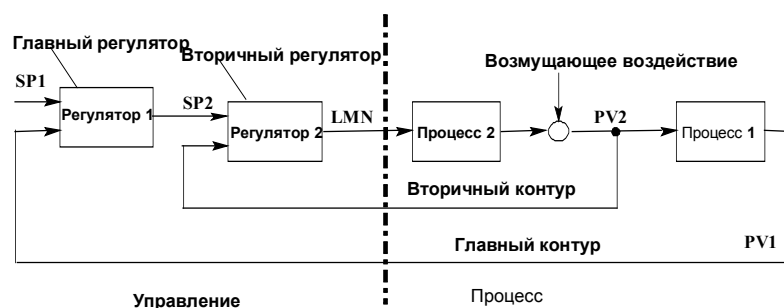
При исполнении программы, управляемой событиями, циклическая программа пользователя прерывается стартовыми событиями (→ классы приоритетов). Если происходит стартовое событие, то исполняемый в данный момент блок прерывается перед следующей командой, и вызывается и исполняется соответствующий организационный блок. Затем циклическая обработка программы продолжается с точки прерывания.

## К

### Каскадное регулирование

Каскадное регулирование включает в себя ряд взаимосвязанных регуляторов, в котором главный регулятор (master) устанавливает задающее воздействие для вторичных (slave) регуляторов в соответствии с текущим сигналом ошибки главной регулируемой величины.

Благодаря включению дополнительных переменных процесса результат регулирования с помощью каскадного регулятора может быть улучшен. Для этого в подходящей точке снимается значение вспомогательной регулируемой величины PV2, которая регулируется в соответствии с эталонным задающим воздействием (выход главного регулятора SP2). Главный регулятор стабилизирует значение переменной процесса PV1 в соответствии с жестко заданным значением SP1 и для этого устанавливает SP2 таким образом, чтобы эта цель могла быть достигнута возможно быстрее и без перерегулирования.



### **Класс приоритета**

Операционная система CPU имеет до 28 классов приоритета, которым ставятся в соответствие различные организационные блоки. Класс приоритета определяет, какие ОВ могут прерывать исполнение других ОВ. Если класс приоритета включает в себя более одного ОВ, то они не прерывают друг друга, а выполняются последовательно.

### **Команда**

Команда (STEP 5 или STEP 7) - это самая малая самостоятельная единица программы, созданной на текстовом языке. Она представляет рабочее предписание для процессора.

### **Коммуникационные SFB для сконфигурированных соединений**

Коммуникационные SFB - это системные функциональные блоки для обмена данными и управления программами.

Примеры обмена данными: SEND, RECEIVE, GET.

Примеры управления программами: перевод CPU партнера по связи в состояние STOP, опрос состояния CPU партнера по связи.

### **Коммуникационные SFC для несконфигурированных соединений**

Коммуникационные SFC – это системные функции для обмена данными и прерывания существующих соединений, установленных с помощью коммуникационных SFC.

### **Константа**

“Константы” - это маркеры для постоянных величин в → логических блоках. Константы применяются для повышения удобочитаемости программы. Например, вместо прямого ввода значения (напр., 10) в → **функциональном блоке** указывается маркер "Max\_iteration\_count" ["Макс\_количество\_итераций"]. Значение константы (например, 10) затем вводится при вызове блока.

### **Контур регулирования**

Под контуром регулирования понимается соединение выхода процесса (регулируемой величины) с входом регулятора и выхода регулятора (управляющего воздействия) с входом процесса, так что регулятор и процесс образуют замкнутый контур.

## Л

### Логический блок

Логический блок в SIMATIC S7 - это блок, содержащий часть программы пользователя STEP 7. Другим типом блока является → **блок данных**, который содержит только данные. Следующий список перечисляет типы логических блоков:

- организационный блок (OB)
- функциональный блок (FB)
- функция (FC)
- системный функциональный блок (SFB)
- системная функция (SFC).

## М

### Меркер (битовая память)

Меркер или битовая память - это память, допускающая побитовый доступ. Основные операции STEP 7 имеют доступ к битовой памяти на запись и чтение (с адресацией побитно, побайтно, пословно и двойными словами). Область битовой памяти может использоваться для сохранения промежуточных результатов.

## О

### Обработка ошибок с помощью OB

Если системная программа распознает определенную ошибку (например, → ошибку доступа в S7), то она вызывает предусмотренный для этого случая организационный блок, в котором программой пользователя может быть установлена реакция CPU на ошибку.

## Объект регулирования

Объектом регулирования (процессом) называется часть установки, в которой регулируемая величина находится под влиянием управляющего воздействия (благодаря изменению энергии, необходимой для перемещения исполнительного органа, или потока материала). Его можно разделить на исполнительное устройство и управляемый процесс.



## Операционная система CPU

Операционная система CPU организует все функции и процессы CPU, не связанные со специальной задачей управления.

## Описание переменной

Описание переменной включает в себя ввод символического имени, типа данных и, возможно, начального значения, адреса и комментария.

## Организационный блок (ОБ)

Организационные блоки образуют интерфейс между операционной системой CPU и программой пользователя. В организационных блоках определяется последовательность обработки программы пользователя.

## Ошибка асинхронная

Асинхронные ошибки – это → **ошибки этапа исполнения программы**, не привязанные к определенному месту в программе пользователя (например, неисправность блока питания, превышение времени цикла). При возникновении такой ошибки операционная система вызывает соответствующий → **организационный блок**, в котором пользователь может запрограммировать реакцию на эту ошибку.

## Ошибка групповая

Групповые ошибки индицируются светодиодами на передней панели модуля в S7-300. При этом светодиоды загораются при появлении ошибок в модуле (внутренних и внешних).

### Ошибки программы пользователя

Ошибки, которые могут возникнуть при обработке программы пользователя в микроконтроллере SIMATIC S7 (в противоположность → **ошибкам процесса**). Обработка ошибок операционной системой производится с помощью ОВ ошибок (система → **классов приоритета**), → **слова состояния** и выходных параметров → **системных функций**.

### Ошибка синхронная

Синхронные ошибки – это ошибки → этапа выполнения программы, которые можно поставить в соответствие определенному месту в программе пользователя (например, ошибка доступа к модулю ввода / вывода). При возникновении такой ошибки операционная система вызывает соответствующий организационный блок, в котором пользователь может запрограммировать реакцию на эту ошибку.

### Ошибка системная

Системные ошибки – это ошибки, происходящие внутри PLC (но не в процессе). Системными ошибками, например, являются программные ошибки в CPU, а также сбои в модулях.

### Ошибки этапа исполнения

Ошибки, возникающие во время обработки программы пользователя в программируемом логическом контроллере (т.е. не в процессе).

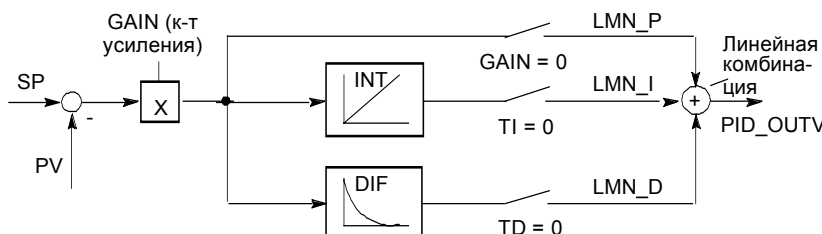
## П

### П–алгоритм

Алгоритм расчета выходного сигнала, в котором имеет место пропорциональная связь между сигналом ошибки и изменением задающего воздействия. Характеристики: установившийся сигнал ошибки, не следует применять на объектах с транспортным запаздыванием.

## Параллельная структура

Параллельная структура - это специальный вид обработки сигналов в регуляторе (математическая обработка). Пропорциональная, интегральная и дифференциальная компоненты рассчитываются параллельно, не взаимодействуя друг с другом, а затем суммируются.



## Параметр

1. Параметр – это переменная логического блока S7

(→ **параметр блока**

→ **фактический параметр**

→ **формальный параметр**).

2. Переменная для настройки поведения модуля (одна или несколько на модуль)

Каждый конфигурируемый модуль при поставке с завода имеет некоторую рациональную базовую установку, которая может быть изменена с помощью STEP 7.

Имеется два вида параметров: статические и динамические

→ **параметр статический** / → **параметр динамический**).

## Параметр, сопутствующий сообщению

Вместе с выводимым сообщением система может выдавать некоторый параметр, который своим значением информирует пользователя о значении переменной или об адресе, которые генерируются в одно время с сообщением.

## Параметр, динамический

Динамические параметры модулей, в отличие от статических, могут быть изменены программой пользователя во время работы модуля с помощью вызова SFC, например, граничные значения аналогового модуля.

## Параметр, статический

Статические параметры модулей, в отличие от динамических, не могут быть изменены программой пользователя, а только через STEP 7, например, задержка входа цифрового модуля ввода.

## Параметры модуля

Параметры модуля – это величины, с помощью которых можно установить режим работы модуля. В зависимости от конкретного модуля некоторые из этих параметров могут быть изменены в программе пользователя.

## Параметры регулятора

Параметры регулятора – это характеристические значения для согласования статических и динамических свойств регулятора с заданными свойствами объекта регулирования или процесса.

## Переменная

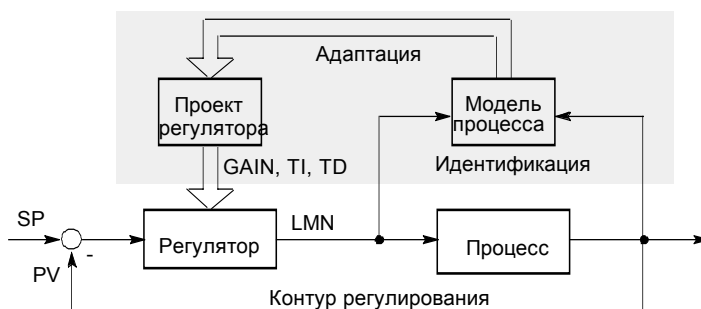
Переменная – это элемент данных с переменным содержимым, который может быть применен в программе пользователя STEP 7. Переменная состоит из адреса (например, M 3.1) и типа данных (например, BOOL) и представляется символом (напр., MOTOR\_ON).

## ПИ–алгоритм

Алгоритм расчета выходного сигнала, в котором изменение управляющего воздействия составляется из компонента, пропорционального сигналу ошибки, и интегрального компонента, значение которого пропорционально сигналу ошибки и времени. Характеристики: отсутствие установившегося сигнала ошибки, более быстрая компенсация, чем в случае И-алгоритма, пригодность для любых процессов.

## ПИД–алгоритм

Алгоритм расчета выходного сигнала, который образуется путем умножения, интегрирования и дифференцирования из сигнала ошибки. ПИД-алгоритм – это параллельная структура. Характеристика: высокое качество регулирования достигается, если транспортное запаздывание объекта регулирования не превышает остальных постоянных времени.



## Подномер (вспомогательный номер)

Номер контролируемого сигнала, если блок, формирующий сообщение, может вести мониторинг более, чем одного сигнала.



### **Полный рестарт**

При запуске CPU (например, при переводе переключателя режимов работы из STOP в RUN или при включении питающего напряжения) перед циклической обработкой программы (OB1) сначала выполняется организационный блок OB101 (рестарт; только у S7-400) или OB100 (полный рестарт). При полном рестарте считывается таблица входов образа процесса и программа пользователя STEP 7 обрабатывается начиная с первой команды в OB1.

### **Прерывание**

Система классов приоритета → SIMATIC S7 распознает 10 различных классов приоритета, управляющих обработкой программы пользователя. К этим классам приоритета относятся, в частности, прерывания, например, аппаратные прерывания. При возникновении прерывания операционная система автоматически вызывает соответствующий организационный блок, в котором пользователь может запрограммировать желаемую реакцию (например, в функциональном блоке).

### **Прерывание аппаратное**

Аппаратное прерывание вызываются модулями, которые обладают способностью к генерации прерываний при возникновении особых событий в процессе. Аппаратные прерывания поступают в CPU, после чего вызывается для обработки соответствующий приоритету прерывания организационный блок.

### **Прерывание по времени**

Прерывание по времени относится к одному из классов приоритета при исполнении программы SIMATIC S7. Оно генерируется в определенный день и момент времени (например, 9:50 или ежечасно, ежеминутно). Затем выполняется соответствующий организационный блок.

### **Прерывание с задержкой**

Прерывание с задержкой относится к одному из классов приоритета при исполнении программы SIMATIC S7. Оно генерируется при завершении работы запущенного в программе пользователя таймера. Затем выполняется соответствующий организационный блок.

### **Приоритет**

Назначая приоритет организационному блоку, Вы определяете возможность прерывания активной в данный момент программы пользователя так, что события с высоким приоритетом прерывают события с более низким приоритетом.

## Приоритет ОВ

→ **Операционная система** CPU различает классы приоритета, например, циклическую обработку программы, обработку программы, управляемую аппаратными прерываниями. Каждому классу приоритета ставится в соответствие → организационный блок (ОВ), в котором пользователь S7 может запрограммировать реакцию на событие. ОВ имеют различные приоритеты, позволяющие обрабатывать их в правильной последовательности при одновременном вызове и дающие возможность ОВ с более высоким приоритетом прерывать ОВ с более низким приоритетом. Стандартные приоритеты могут изменяться пользователем.

## Программа пользователя

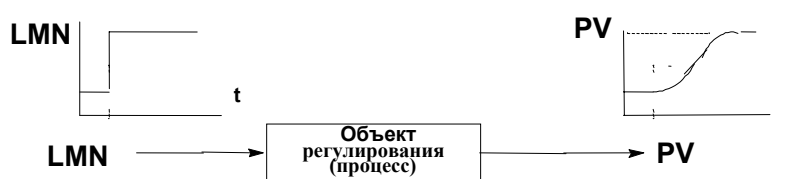
Программа пользователя содержит все операторы и описания, а также данные для обработки сигналов, с помощью которых можно управлять системой или процессом. Она ставится в соответствие программируемому модулю (→ **Модуль программируемый**) и может быть структурирована делением на более мелкие единицы, известные как → блоки.

## Пропорциональное исполнительное звено (привод)

→ **Широтно-импульсная модуляция**

## Процесс

→ Объектом регулирования (процессом) называется часть установки, в которой регулируемая величина находится под влиянием управляющего воздействия (благодаря изменению энергии, необходимой для перемещения исполнительного органа, или потока материала). Его можно разделить на исполнительное устройство и управляемый процесс.



**Р**

**Реакция на ошибку**

Реакция на ошибку → этапа исполнения программы. Операционная система может реагировать следующим образом: переводом контроллера в состояние STOP, вызовом организационного блока, в котором пользователь может запрограммировать реакцию на ошибку, или отображением ошибки.

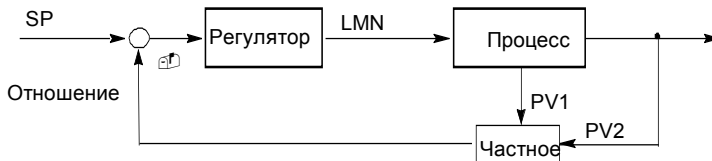
**Результат логической операции (RLO, VKE)**

Результат логической операции (англ. RLO, нем. VKE) – это текущее состояние сигнала в процессоре, применяемое для дальнейшей двоичной обработки сигнала. Определенные операции выполняются или не выполняются в зависимости от предшествующего состояния RLO.

**Регулирование пропорциональное**

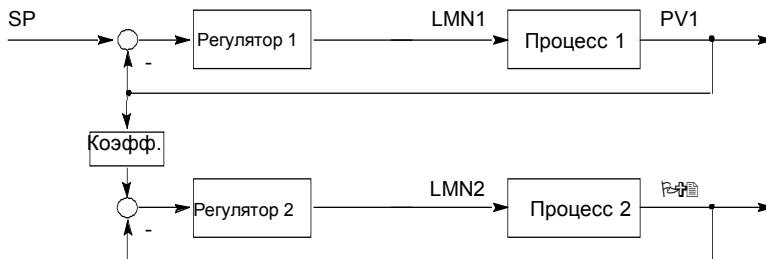
- Одноконтурный пропорциональный регулятор

Одноконтурный регулятор отношения используется тогда, когда отношение двух регулируемых величин важнее, чем их абсолютные значения.



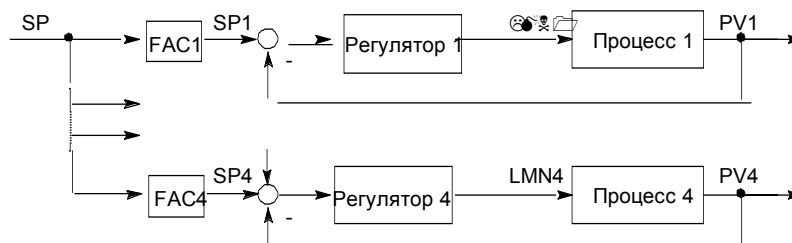
- Многоконтурный пропорциональный регулятор

В двухконтурном регуляторе отношения выдерживается постоянным отношение двух регулируемых величин PV1 и PV2. Для этого задающее значение 2-го контура регулирования вычисляется из регулируемой величины 1-го контура. Это отношение поддерживается даже при динамическом изменении регулируемой величины PV1.



## Регулирование смешанное

Смешанное регулирование - это регулирующая структура, в которой задающее воздействие для общего количества SP преобразуется в процентное содержание отдельных компонентов. Сумма коэффициентов смешивания FAC должна быть равна 1 (=100 %).



## Регулятор

Регулятор – это устройство, в котором непрерывно рассчитывается сигнал ошибки и генерируется управляющее воздействие с целью быстро и без перерегулирования устранить сигнал ошибки.

## Регулятор непрерывного управления

Регулятор непрерывного управления – это регулятор, в котором каждое изменение сигнала ошибки приводит к изменению управляющей переменной, которая может принимать любое значение из разрешенного для нее диапазона.

## Регулятор пошагового управления

Регулятор пошагового управления – это регулятор квазинепрерывного управления с дискретным выходом (для привода, управляемого мотором, и имеющего И-составляющую). Привод имеет трехуровневую характеристику управления, например: вверх – стоп – вниз (или открыть – фиксировать – закрыть). (-> **Трехпозиционный регулятор**).

## Рестарт

При запуске CPU (например, при переводе переключателя режимов работы из STOP в RUN или при включении напряжения сети) перед циклической обработкой программы (OB1) сначала выполняется организационный блок OB100 (полный рестарт) или организационный блок OB101 (рестарт; только у S7-400). При рестарте считывается таблица входов образа процесса, и обработка программы пользователя продолжается с того места, на котором она была прервана при последнем останове (STOP, отключение питания).

## С

### **Связь, двусторонняя**

При использовании для обмена данными коммуникационных SFB различают одностороннюю и двустороннюю связь. Связь является двусторонней, если SFB имеется как на локальном, так и на удаленном модуле, например, коммуникационные SFB "USEND" и "URCV".

### **Связь, односторонняя**

При использовании для обмена данными коммуникационных SFB различают одностороннюю и двустороннюю связь. Связь является односторонней, если SFB имеется только на локальном модуле, например, SFB "GET".

### **Связь, односторонняя**

При использовании для обмена данными коммуникационных SFB различают одностороннюю и двустороннюю связь. Связь является двусторонней, если коммуникационные SFB имеются как в локальном, так и в удаленном модулях, например, коммуникационные SFB "USEND" и "URCV".

### **Система автоматического регулирования**

Система автоматического регулирования – это устройство, в котором непрерывно рассчитывается сигнал ошибки и генерируется управляющее воздействие с целью быстро и без перерегулирования устранить сигнал ошибки.

### **Символическое программирование**

Язык программирования STEP 7 дает возможность применять символические обозначения вместо адресов STEP 7. Это значит, например, что адрес "Q 1.1" может быть заменен символическим именем "Valve 17".

Так называемый список (таблица) символов в STEP 7 создает при этом связь между → **адресом** и соответствующим символическим именем.

### **Системная ошибка**

Системные ошибки – это ошибки, которые могут возникать внутри контроллера (т.е. не в процессе). Системными ошибками могут быть, например, программные ошибки в CPU и неисправности в модулях.

### **Системная функция (SFC)**

Системная функция (SFC) – это встроенная в операционную систему CPU → функция, которая при необходимости может быть вызвана в программе пользователя STEP 7.

## **Системный функциональный блок (SFB)**

Системный функциональный блок (SFB) – это встроенный в операционную систему CPU → функциональный блок, который может быть при необходимости вызван в программе пользователя STEP 7.

## **Сообщение**

Сообщение – это "доклад", который выдается при свершении события. Сообщение может быть выдано на соответствующий сконфигурированный дисплей и при этом оно может содержать информацию, такую как приоритет, локализация, время свершения, а также тип (входное/выходное).

## **Сообщение, связанное с блоком**

Сообщение, которое конфигурируется для блока (FB или DB), способного генерировать сообщения.

## **Сообщение, связанное с символом (символическим именем)**

Сообщение, которое конфигурируется для входа, выхода, меркера, блока данных в таблице символов. Во время конфигурирования для функции SCAN задается временной интервал для мониторинга сигнала.

## **Список мнемоник**

Список мнемоник или операторов (английское сокращение STL, немецкое сокращение AWL) является языком ассемблера → **STEP 7**. Если программа написана на STL, то отдельные команды соответствуют рабочим шагам, выполняемым CPU при обработке программы.

## **Стандартная функция**

Стандартные функции – это поставляемые фирмой SIEMENS функции для решения сложных задач.

## **Стандартный функциональный блок**

Стандартные функциональные блоки – это поставляемые фирмой SIEMENS функциональные блоки для решения сложных задач.

## **Стартовая информация**

Когда операционная система вызывает организационный блок, она передает стартовую информацию, которая может быть проанализирована в программе пользователя.

### **Стартовое событие**

Стартовые события – это определенные события, например, ошибки или прерывания, побуждающие операционную систему вызывать подходящий организационный блок.

## **Т**

### **Трехпозиционный регулятор**

Регулятор, который может принимать только три дискретных состояния; например, “нагрев - выключено - охлаждение” или “направо - остановка - налево” (-> **Регулятор пошагового управления**).

## **У**

### **Удаленное устройство**

Удаленные устройства – это устройства, например, принтеры или компьютеры, доступ к которым осуществляется через сеть. В отличие от локальных устройств, при их установке им должен быть присвоен сетевой адрес.

### **Устройство управления**

Устройство управления – это законченное устройство, использующее переменную процесса. Оно состоит из контроллера, управляемого устройства (привода) и датчика (датчиков) для измерения параметров процесса.

### **Устройство отображения**

Устройство отображения – это устройство, используемое для отображения результатов процесса.

## **Ф**

### **Фактические параметры**

Фактические параметры заменяют формальные параметры при вызове функционального блока (FB) или функции (FC), например, формальный параметр “REQ” заменяется фактическим параметром “I 3.6”.

## Формальные параметры

Формальный параметр – это метка-заполнитель для фактического параметра в параметризуемых логических блоках. В FB и FC формальные параметры описываются пользователем, в SFB и SFC они уже имеются. При вызове блока формальному параметру ставится в соответствие фактический параметр, так что вызванный блок работает с этим фактическим значением. Формальные параметры относятся к  
→ локальным данным блока и делятся на входные, выходные и проходные (in/out) параметры.

## Ш

### Широтно-импульсная модуляция

Широтно-импульсная модуляция – это способ влияния на управляющее воздействие при дискретном выходе. Расчетное управляющее воздействие в процентах преобразуется в пропорциональную сигналу длительность импульса (ED), на выходе управляющего воздействия, например,

$$100 \% T_p = T_A \text{ или } = \text{CYCLE.}$$

## Я

### Язык программирования STEP 7

Язык программирования для контроллеров SIMATIC S7. Программист может применять STEP 7 в различных формах представления:

- список мнемоник (операторов),
- функциональный план,
- контактный план.



## **O**

### **OB запуска**

В зависимости от положения переключателя видов запуска (только в S7-400), причины запуска (восстановление исчезнувшего напряжения питания; ручное переключение из STOP в RUN переключателем режимов работы или командой с устройства программирования) операционной системой вызывается организационный блок запуска “Полный рестарт” или “Рестарт” (имеется только в S7-400). В OB запуска пользователь SIMATIC S7 может запрограммировать, например, как система будет запускаться вновь после исчезновения напряжения.

### **OB ошибок (обработки ошибок)**

OB обработки ошибок – это организационные блоки, с помощью которых пользователь может запрограммировать реакцию на ошибку. Правда, запрограммированная реакция возможна только тогда, когда ошибка не приводит к остановке контроллера. Соответствующие OB ошибок имеются для различных типов ошибок (например, OB ошибок адресации, OB ошибок доступа в S7).

## **OB1**

Организационный блок OB1 – это интерфейс пользователя с системной программой для циклической обработки программы.

## **S**

### **SCAN**

Функция операционной системы, встроенная в CPU, которая используется для сканирования с целью обнаружения сигналов в установленные интервалы времени и определения изменения этих сигналов.

### **STEP 7**

Программное обеспечение для разработки программ пользователя для контроллеров SIMATIC S7.

## **STL**

→ **Список мнемоник**

**V**

**VKE**

Результат логической операции (англ. RLO, нем. VKE) – это текущее состояние сигнала в процессоре, применяемое для дальнейшей двоичной обработки сигнала. Определенные операции выполняются или не выполняются в зависимости от предшествующего состояния RLO.  
(-> **Результат логической операции**)

# Предметный указатель

## А

Аппаратное прерывание 1-2  
потеряно 1-33  
Асинхронная ошибка 1-3, 1-10, 12-1  
блокирование с помощью SFC39  
DIS\_IRT 12-2  
задержка с помощью SFC41  
DIS\_AIRT 12-7  
разблокирование с помощью SFC40  
EN\_IRT 12-2, 12-6, 12-6  
разблокирование с помощью SFC42  
EN\_AIRT 12-8

## Б

Батарея разряжена 1-31  
Битовый массив области  
входов/выходов  
сброс с помощью SFC80 14-5  
установка с помощью SFC79 14-4  
Блок данных  
создание с помощью SFC22  
CREAT\_DB 3-9  
удаление с помощью SFC23 3-11  
Блок скомпонованного кода, вызов 29-2

## В

Ведомые часы  
синхронизация 5-4  
Ведущие часы 5-2  
Внешняя ошибка 25-4, 25-16, 25-32  
Внутренняя ошибка 1-32, 2-5, 8-14  
Временные переменные (TEMP)

требуемые для ОВ 1-5, 1-9, 1-11, 13-3  
Время задержки 1-10  
Время суток (TOD) 5-2  
Встроенная функция измерения  
частоты 25-40  
Встроенная функция счетчика 25-34  
Вызов блока скомпонованного кода 29-2

## Г

Генератор последовательностей  
реализация 14-6  
Главный программный цикл (ОВ1) 1-4  
Группы ведомых DP-устройств  
синхронизация 16-5

## Д

Дата 5-2  
Демаскирование  
событий, связанных с ошибками 11-2  
Диагностическая информация модуля  
16-17, 31-5  
Диагностические  
данные 7-3, 13-9, 16-17, 30-1  
содержимое 8-9, 13-14  
структура 13-14  
Диагностические данные  
модуля 30-2, 32-18  
Диагностические события 32-18  
Диагностический буфер 1-34, 11-2, 13-11  
Диагностическое прерывание 12-3, 30-3

### З

Задержка включения  
генерирование 22-4  
Задержка выключения  
генерирование 22-6  
Задержка программы пользователя  
с помощью SFC47 WAIT 4-3  
Заменяющее значение  
запись в ACCU 1 с помощью  
SFC44 REPL\_VAL 3-16  
Запись данных  
запись 7-4  
запись с помощью  
SFC58 WR\_REC 7-12  
чтение 7-4  
чтение с помощью SFC59 RD\_REC 7-13  
Запись  
консистентных данных в стандартное  
ведомое DP-устройство 16-8  
Запись данных в удаленный CPU  
с помощью SFB15 19-27  
Запуск 1-41

### И

Импульс  
генерирование 22-2  
Инициализация области памяти  
с помощью SFC21 FILL 3-8, 33-2  
Информация об ошибках 2-2  
общая 2-4  
специфическая 2-4  
SFC22 CREAT\_DB 3-9  
SFC23 DEL\_DB 3-11  
SFC34 QRY\_DINT 10-5  
SFC40 EN\_IRT 12-6  
SFC42 EN\_AIRT 12-8

### К

Класс приоритета 1-1, 1-5, 1-9, 1-11,  
1-13, 1-14, 1-15, 1-17, 1-18, 1-20, 1-28,  
1-23, 1-24, 1-26, 1-29, 1-30, 1-31, 1-34,  
1-37, 1-40, 1-42, 1-46, 1-48, 10-14, 12-6  
OB1 1-4  
OB121 1-51  
OB122 1-54  
OB20 1-10

OB35 1-12  
OB80 1-28  
OB81 1-30  
OB82 1-32  
OB85 1-37  
Коммуникации  
данные о состоянии 31-19  
Коммуникационные события 32-2  
Коммуникационные SFB См. SFB  
Коммуникационные SFC 18-12  
Коммуникационные SFC для  
несконфигурированных соединений S7  
18-2, 20-1  
Контроль времени цикла 30-3  
Копирование переменных  
с помощью SFC20 BLKMOV 3-2

### М

Маскирование  
ошибок 11-2

### Н

Некоординированная передача данных  
с помощью SFB8 19-12  
Некоординированный прием данных  
с помощью SFB9 21-16

### О

Области памяти 3-5  
Обновление таблицы входов образа  
процесса 14-2  
Обновление таблицы выходов образа  
процесса 14-3  
Обработка ошибок 11-2  
Обратный счет 22-9  
Обрыв провода 30-5, 30-6  
Организационные блоки (OB)  
обзор 1-1  
OB аппаратных ошибок CPU  
(OB84) 1-36  
OB диагностических прерываний  
(OB82) 1-32  
OB запуска

(OB100, OB101, OB102) 1-47  
ОВ коммуникационных ошибок  
(OB87) 1-43  
ОВ неисправности стойки  
(OB86) 1-40  
ОВ ошибок класса приоритета  
(OB85) 1-37  
ОВ полного рестарта  
(OB100) 1-47  
ОВ прерываний по времени  
(OB 10 ... 17) 1-6  
ОВ прерываний при установке/ удалении  
модуля (OB83) 1-34  
ОВ рестарта (OB101) 1-47  
Организационный блок (OB) 1-1  
фоновый ОВ (OB90) 1-45  
ОВ мультипроцессорных прерываний  
(OB60) 1-19  
OB1 1-4  
OB121 1-51  
OB122 1-54  
OB20 1-10  
OB35 1-12  
OB80 1-28  
OB81 1-24  
OB82 1-32  
OB85 1-37  
Ошибка  
асинхронная 12-1  
АЦП/ЦАП 30-3  
маскирование 11-2  
ОЗУ 31-45  
синхронная 2-4, 1-51, 1-54  
программирования 1-51  
СППЗУ 31-45  
выравнивания  
при записи 1-53  
при чтении 1-53  
Ошибка доступа  
к входам/выходам 1-3

## П

Пакет GD  
выборка с помощью SFC61 17-4  
передача с помощью SFC60 17-2  
Память пользователя  
сжатие с помощью SFC25 3-14  
Передача параметров  
с помощью SFC55 WR\_PARM 7-6  
с помощью SFC56 WR\_DPARM 7-8

Передача сегментированных данных  
с помощью SFB12 19-21  
Плата памяти 30-3  
Повторный запуск контроля времени  
цикла с помощью  
SFC43 RE\_TRIGR 1-4, 4-1  
Прерывание 12-2  
блокирование  
с помощью SFC39 DIS\_IRT 12-2, 12-6  
деблокирование  
с помощью SFC40 EN\_IRT 12-6, 12-6  
деблокирование  
с помощью SFC42 EN\_AIRT 12-7, 12-8  
задержка  
с помощью SFC41 DIS\_AIRT 12-2, 12-7  
классы 12-2  
с задержкой (OB20) 1-10  
циклическое (OB35) 1-12  
Прерывание по времени 1-2, 9-7, 9-8  
активация  
с помощью SFC30 ACT\_TINT 9-7  
опрос  
с помощью SFC31 QRY\_TINT 9-8  
отмена  
с помощью SFC29 CAN\_TINT 1-8, 10-6  
теплый рестарт 9-3  
условия для вызова 9-2  
установка с помощью SFC28  
холодный рестарт 9-3  
OB 9-2  
SET\_TINT 8-5  
Прерывание по ошибке  
асинхронной 12-2  
синхронной 12-2  
Прерывание  
при удалении/вставке модуля 11-3  
Прерывание с задержкой 9-2, 12-2  
запрос  
с помощью SFC34 QRY\_DINT 9-5  
запуск  
с помощью SFC32 SRT\_DINT 9-4  
отмена  
с помощью SFC33 CAN\_DINT 10-6  
условия для вызова 9-2  
Прерывания  
ОВ аппаратных прерываний 1-14  
Прерывания  
по асинхронным ошибкам 1-3  
Прерывания  
по синхронным ошибкам 1-3  
Прерывания  
с задержкой OB20 1-10  
Прием сегментированных данных

с помощью SFB13 19-25  
 Программа  
 циклическая 1-4  
 Программирование  
 OB1 1-4  
 OB121 1-51  
 OB122 1-54  
 OB20 1-10  
 OB35 1-12  
 OB80 1-28  
 OB81 1-24  
 OB82 1-32  
 OB85 1-37  
 Прямой счет 22-8  
 Прямой и обратный счет 22-10

**Р**

Регистр ошибок 10-2  
 чтение  
 с помощью SFC38 READ\_ERR 10-14  
 Регулирование  
 непрерывное  
 с помощью SFB41 24-2  
 пошаговое с помощью SFB42 24-10  
 Рестарт 1-41

**С**

Синфазная ошибка  
 аналоговый модуль ввода 30-5  
 аналоговый модуль вывода 30-6  
 Синхронизация  
 многопроцессорного режима 1-47  
 часов 5-4  
 Синхронные ошибки 12-2  
 демаскирование  
 с помощью SFC37 DMSK\_FLT 11-13  
 маскирование  
 с помощью SFC36 MSK\_FLT 11-12  
 OB121 1-51  
 OB122 1-54  
 Системные данные 31-2  
 Системные области 31-10  
 Слот модуля  
 опрос логического адреса 15-2  
 Событие 32-1  
 идентификатор (ID) 12-11, 32-2  
 класс 32 -2

События, связанные  
 со стандартными OB 32 -2  
 События, связанные  
 с состоянием STOP 32 -2  
 События H/F 32 -2  
 Состояние прерывания 31-17  
 Состояние удаленного партнера  
 запрос 19-46  
 получение 19-48  
 Список состояний системы 31-2  
 подписки 31-5  
 Стандартные  
 события пользователя 32-20  
 Счетчик (CPU 312) 24-2  
 Счетчики  
 ошибка номера 10-9  
 Счетчик рабочего времени 6-2  
 диапазон значений 6-2  
 запуск  
 с помощью SFC3 CTRL\_RTM 6-4  
 останов  
 с помощью SFC3 CTRL\_RTM 6-4  
 считывание  
 с помощью SFC4 READ\_RTM 6-5  
 установка  
 с помощью SFC2 SET\_RTM 6-3  
 характеристики 6-2  
 Считывание  
 диагностических данных  
 ведомые DP-устройства 16-17  
 консистентных данных  
 ведомые DP-устройства 16-20  
 с помощью SFC51 RDSYSST  
 Считывание времени  
 с помощью SFC1 READ\_CLK 5-3  
 Считывание данных из удаленного CPU  
 с помощью SFB14 19-30  
 Считывание системного времени  
 с помощью SFC64 TIME\_TCK 6-6  
 Считывание состояния системы  
 с помощью SFC51 RDSYSST 13-6  
 Считывание стартовой информации OB  
 с помощью SFC6 12-3

**Т**

Таблица описания переменных  
 для OB1 1-4  
 для OB10 1-6  
 для OB121 1-51  
 для OB122 1-54

для OB20 1-10  
для OB35 1-12  
для OB80 1-28  
Типы блоков 3-8

## У

Управление функционированием  
в H-системах с помощью SFC90  
Установка времени  
с помощью SFC0 SET\_CLK 5-2

## Ф

Фильтрация  
ошибок 10-3  
Фильтр ошибок  
ошибки доступа 11-3  
ошибки программирования 11-3, 11-4  
Фильтр ошибок доступа 11-3, 11-5  
Фильтр ошибок доступа  
для CPU 417 и CPU 417H 11-6  
Фильтр ошибок программирования 11-4  
Функции IEC  
обзор 23-2  
требования к рабочей и  
загрузочной памяти 23-3

## Ц

Циклические прерывания 12-2  
OB35 1-12

## Ч

Частотомер (CPU 312) 25-40  
Часы  
ведущие 5-2  
синхронизация 5-2

## Ш

Широтно-импульсная модуляция 22-15

Системные и стандартные функции для S7-300 и S7-400  
A5E00261410-01

**A**

ACCFLT\_... 11-5  
 ACCFLT\_MASKED 11-12  
 ACCFLT\_QUERY 11-14  
 ACCFLT\_RESET\_MASK 11-13  
 ACCFLT\_SET\_MASK 11-12  
 ACT\_TINT 9-7  
 ALARM 21-8  
 ALARM\_8 21-13  
 ALARM\_8P 21-10  
 ALARM\_S 21-25  
 ALARM\_SC 21-28  
 ALARM\_SQ 21-25  
 AR\_SEND 21-16

**B**

BLK 3-8  
 BLKMOV 3-2  
 BRCV 19-23  
 BSEND 17-12  
 BVAL 3-7

**C**

CAN\_DINT 10-6  
 CAN\_TINT 1-8, 10-6  
 CDT 5-3  
 COMPRESS 3-14  
 CONT\_C 24-2  
 CONT\_S 24-10  
 CONTROL 19-51  
 COUNT 25-34  
 CPU  
 переход в состояние STOP  
 с помощью SFC46 STP 4-4  
 характеристики 31-7  
 CQ 6-5  
 CREATE\_DB 3-9  
 CTD 22-9  
 CTRL\_RTM 6-4  
 CTU 22-8  
 CTUD 24-10  
 CV 6-5

**D**

DB\_NUMBER 3-9  
 DEL\_DB 3-11  
 DIS\_AIRT 12-2, 12-7  
 DIS\_IRT 12-2, 12-6  
 DIS\_MSG 21-18  
 DMSK\_FLT 11-13  
 DP\_PRAL 16-2  
 DPNRM\_DG 16-17  
 DPRD\_DAT 16-20  
 DPSYC\_FR 16-5  
 DPWR\_DAT 16-22  
 DRUM 14-6  
 DSTBLK 2-7, 3-6  
 DTIME 10-4

**E**

EN\_AIRT 12-7, 12-8  
 EN\_IRT 12-6  
 EN\_MSG 21-20  
 EVENTN 13-12, 13-13

**F**

FC1 23-5  
 FC10 23-11  
 FC11 23-18  
 FC12 23-9  
 FC13 23-11  
 FC14 23-9  
 FC15 23-12  
 FC16 23-22  
 FC17 23-18  
 FC18 23-10  
 FC19 23-12  
 FC2 23-17  
 FC20 23-19  
 FC21 23-19  
 FC22 23-14  
 FC23 23-10  
 FC24 23-13  
 FC25 23-14  
 FC26 23-20  
 FC27 23-15  
 FC28 23-10  
 FC29 23-13



FC3 23-5  
FC30 23-22  
FC31 23-20  
FC32 23-21  
FC33 23-7  
FC34 23-7  
FC35 23-7  
FC36 23-16  
FC37 23-23  
FC38 23-23  
FC39 23-24  
FC4 23-17  
FC40 23-8  
FC5 23-22  
FC6 23-6  
FC7 23-6  
FC8 23-6  
FC9 23-9  
FILL 3-7

## G

GADR\_LGC 15-2  
GD\_RCV 17-4  
GD\_SND 17-2  
GET 19-30

## H

H\_CTRL 26-2

## I

I\_ABORT 20-23  
I\_GET 20-19  
I\_PUT 20-21  
INFO1 13-12, 13-13  
INFO2 13-12, 13-13  
IOID 2-7, 7-4

## L

LADDR 2-7, 7-4  
LGC\_GADR 15-4  
LOW\_LIMIT 25-34

## M

MODE 2-7, 5-5, 12-2, 12-4  
MP\_ALM 4-6  
MSK\_FLT 11-12

## N

NOTIFY 21-5  
NR 6-3, 6-4, 6-5

## O

OB1 1-4  
OB20 1-10  
OB35 1-12  
OB80 1-28  
OB81 1-24  
OB82 1-32  
OB85 1-37  
OB ошибок класса приоритета 1-26  
OB\_NR 9-5  
OB100, OB101 и OB102 1-47  
OB121 1-51  
OB10 ... OB17 1-6  
OB20 ... OB23 1-10  
OB30 ... OB38 1-12  
OB40 ... OB47 1-14  
OB55 1-16  
OB56 1-17  
OB57 1-18  
OB60 1-19  
OB61 1-21  
OB70 1-22  
OB72 1-24  
OB73 1-26  
OB80 1-28  
OB81 1-30  
OB82 1-32  
OB83 1-34  
OB84 1-36  
OB85 1-37  
OB86 1-40  
OB87 1-43  
OB90 1-45  
OB100 1-47  
OB101 1-47  
OB102 1-47  
OB121 1-51

OB122 1-54

**P**

PARM\_MOD 7-9  
 PDT 5-2  
 PERIOD 8-5  
 PRGFLT\_ESR 10-14  
 PRGFLT\_MASKED 11-12, 11-13  
 PRGFLT\_QUERY 10-14  
 PRGFLT\_RESET\_MASK 11-13  
 PRGFLT\_SET\_MASK 11-12  
 PRINT 17-20  
 PULSEGEN 22-15  
 PUT 17-18  
 PV 6-3

**Q**

QRY\_DINT 10-5  
 QRY\_TINT 9-8

**R**

RD\_LGADR 14-6  
 RD\_REC 7-12  
 RD\_SINFO 12-3  
 RDSYSST 13-6, 31-2  
 RE\_TRIGR 1-4, 4-1  
 READ\_CLK 5-3  
 READ\_ERR 10-14  
 READ\_RTM 6-5  
 RECNUM 7-6, 7-9, 7-11, 7-13  
 RECORD 7-6, 7-11  
 REPL\_VAL 3-14  
 RESUME 17-30

**S**

S 6-4  
 SDT 8-5  
 SEND 13-12, 13-13  
 SET 13-4, 13-5  
 SET\_CLK 5-2  
 SET\_RTM 6-3

SET\_TINT 8-5  
 SFB0 CTU 22-8  
 SFB1 CTD 22-9  
 SFB12 BSEND 19-21  
 SFB13 BRCV 19-25  
 SFB14 GET 19-31  
 SFB15 PUT 19-27  
 SFB16 PRINT 17-20  
 SFB19 START 17-26  
 SFB2 CTUD 24-10  
 SFB20 STOP 17-28  
 SFB21 RESUME 17-30  
 SFB22 STATUS 19-46  
 SFB23 USTATUS 19-48  
 SFB3 TP 20-2  
 SFB32 DRUM 13-6  
 SFB33 ALARM 19-7  
 SFB34 ALARM\_8 19-13  
 SFB35 ALARM\_8P 19-10  
 SFB36 NOTIFY 19-5  
 SFB37 AR\_SEND 19-15  
 SFB4 TON 20-4  
 SFB41 CONT\_C 24-2  
 SFB42 CONT\_S 24-10  
 SFB43 PULSEGEN 22-15  
 SFB5 TOF 20-6  
 SFB8 USEND 19-12  
 SFB9 URCV 19-15  
 SFC0 SET\_CLK 5-2  
 SFC1 READ\_CLK 5-3  
 SFC10 DIS\_MSG 21-18  
 SFC11 DPSYC\_FR 16-5  
 SFC13 DPNRM\_DG 16-17  
 SFC14 DPRD\_DAT 16-20  
 SFC15 DPWR\_DAT 16-22  
 SFC17 ALARM\_SQ 21-27  
 SFC18 ALARM\_S 21-27  
 SFC19 ALARM\_SC 21-30  
 SFC2 SET\_RTM 6-3  
 SFC20 BLKMOV 3-2  
 SFC21 FILL 3-7  
 SFC22 CREAT\_DB 3-9  
 SFC23 DEL\_DB 3-11  
 SFC25 COMPRESS 3-14  
 SFC26 UPDAT\_PI 14-2  
 SFC27 UPDAT\_PO 14-3  
 SFC28 SET\_TINT 9-5  
 SFC29 CAN\_TINT 9-6  
 SFC3 CTRL\_RTM 6-4  
 SFC30 ACT\_TINT 9-7  
 SFC31 QRY\_TINT 9-8  
 SFC32 SRT\_DINT 10-4  
 SFC33 CAN\_DINT 10-6

SFC34 QRY\_DINT 10-5  
 SFC35 MP\_ALM 4-6  
 SFC36 MSK\_FLT 11-12  
 SFC37 DMSK\_FLT 11-13  
 SFC38 READ\_ERR 11-14  
 SFC39 DIS\_IRT 12-4  
 SFC4 READ\_RTM 6-5  
 SFC40 EN\_IRT 12-6  
 SFC41 DIS\_AIRT 12-7  
 SFC42 EN\_AIRT 12-8  
 SFC43 RE\_TRIGR 4-3  
 SFC44 REPL\_VAL 3-16  
 SFC46 STP 4-4  
 SFC46 TIME\_TCK 6-6  
 SFC47 WAIT 4-5  
 SFC48 SNC\_RTCB 5-4  
 SFC49 LGC\_GADR 15-4  
 SFC5 GADR\_LGC 15-2  
 SFC50 RD\_LGADR 15-6  
 SFC51 RDSYSST 13-6  
 SFC52 WR\_USMSG 13-11  
 SFC55 WR\_PARM 7-6  
 SFC56 WR\_DPARM 7-8  
 SFC57 PARM\_MOD 7-9  
 SFC58 WR\_REC 7-11  
 SFC59 RD\_REC 7-12  
 SFC6 RD\_SINFO 13-3  
 SFC60 GD\_SND 17-2  
 SFC61 GD\_RCV 17-4  
 SFC62 CONTROL 19-51  
 SFC64 TIME\_TCK 6-6  
 SFC65 X\_SEND 20-8  
 SFC66 X\_RCV 20-10  
 SFC67 X\_GET 20-16  
 SFC68 X\_PUT 20-14  
 SFC69 X\_ABORT 20-18  
 SFC7 DP\_PRAL 16-2  
 SFC72 I\_GET 20-19  
 SFC73 I\_PUT 20-21  
 SFC74 I\_ABORT 20-23  
 SFC79 SET 14-4  
 SFC9 EN\_MSG 21-20  
 SFC90 H\_CTRL 26-2  
 SNC\_RTCB 5-4  
 SRCBLK 2-7, 3-6  
 SRT\_DINT 10-4  
 SSL\_HEADER 13-6  
 SSL-ID 31-2  
 START 19-39  
 STATUS 19-46  
 STEP 7 1-1  
 STOP 19-42

**T**

TIME\_TCK 6-6  
 TOF 22-6  
 TON 22-4  
 TP 22-2

**U**

UP\_LIMIT 2-7  
 UPDAT\_PI 14-2  
 UPDAT\_PO 14-3  
 URCV 19-15  
 USEND 19-12  
 USTATUS 19-48

**W**

WAIT 4-5  
 WR\_DPARM 7-8  
 WR\_PARM 7-6  
 WR\_REC 7-11  
 WR\_USMSG 13-11  
 WT 4-5

**X**

X\_ABORT 20-18  
 X\_GET 20-16  
 X\_PUT 20-14  
 X\_RCV 20-10  
 X\_SEND 20-8



# Литература

- /30/ Getting Started: Working with STEP 7 V5.0  
[Введение: Работа со STEP 7 V5.0]
  
- /70/ Руководство: S7-300 Programmable Controller  
Hardware and Installation  
[Программируемый контроллер S7-300.  
Аппаратура и установка]
  
- /71/ Справочное руководство: S7–300, M7–300 Programmable  
Controllers  
Module Specifications  
[Программируемые контроллеры S7-300, M7-300  
Описания модулей]
  
- /72/ Instructions List: S7–300 Programmable Controller  
[Список команд: Программируемый контроллер S7-300]
  
- /73/ Manual: Programmable Controllers,  
Integrated Functions CPU 312 IFM/314 IFM  
[Руководство: Программируемые контроллеры,  
встроенные функции CPU 312 IFM/314 IFM]
  
- /100/ Руководство: S7-400 and M7-400, Programmable Controllers  
Hardware and Installation  
[Аппаратное обеспечение программируемых контроллеров S7-400,  
M7-400 и их инсталляция]
  
- /101/ Справочное руководство: S7–400, M7–400 Programmable controllers  
Module Specifications  
[Программируемые контроллеры S7-400, M7-400  
Описания модулей]

- /102/ Instructions List: S7-400 Programmable Controller  
[Список команд: Программируемый контроллер S7-400]
- /230/ Руководство по преобразованию: From S5 to S7  
[От S5 к S7]
- /231/ Руководство: Configuring Hardware and Communication Connections,  
STEP 7 V5.0  
[Конфигурирование аппаратуры и коммуникационных связей, STEP  
7 V5.0]
- /232/ Справочное руководство: Statement List (STL) for S7-300 and S7-  
400 [Список операторов (STL) для S7-300 и S7-400]
- /233/ Справочное руководство: Ladder Logic (LAD) for S7-300 and S7-400  
[Контактный план (LAD) для S7-300 и S7-400]
- /234/ Руководство: Programming with STEP 7 V5.0  
[Программирование с помощью STEP 7 V5.0]
- /236/ Справочное руководство: Function Block Diagram (FBD) for S7-300  
and S7-400 [Функциональный план (FBD) для S7-300 и S7-400]
- /250/ Руководство: Structured Control Language (SCL) for S7-300 and S7-  
400 Programming  
[Язык структурного управления (SCL) для программирования  
S7-300 и S7-400]
- /251/ Руководство: S7-GRAPH for S7-300 and S7-400,  
Programming Sequential Control Systems  
[S7-GRAPH для S7-300 и S7-400,  
Программирование систем последовательного управления]
- /252/ Руководство: S7-HiGraph for S7-300 and S7-400,  
Programming State Graphs  
[S7-HiGraph для S7-300 и S7-400, Программирование графов  
состояния]
- /254/ Руководство: CFC Continuous Functions Charts  
[CFC, Схемы последовательных функций]

- /270/ Руководство: S7-PDIAG for S7-300 and S7-400  
"Configuring Process Diagnostics for LAD, STL, and FBD"  
[S7-PDIAG для S7-300 и S7-400  
"Конфигурирование диагностики процесса для контактного плана,  
списка операторов и функционального плана"]
- /350/ Руководство пользователя: SIMATIC 7, Standard Controller  
[SIMATIC7, Стандартный контроллер]

