

Библиотека VIPA SPEED7

OPL_SP7-LIB | SW90AS0MA V10.004 | Руководство

HB00 | OPL_SP7-LIB | SW90AS0MA V10.004 | ru | 18-30

Библиотека функциональных блоков для обмена данными с использованием
протокола Modbus



VIPA GmbH
Омштрассе, 4
91074, Херцогенаурах
Тел.: +49 9132 744-0
Факс: +49 9132 744-1864
Email: info@vipa.com
Internet: www.vipa.com

Содержание

1.	Общие сведения	4
1.1	Авторские права	4
1.2	Информация о руководстве	5
2.	Важные замечания	6
2.1	Общая информация	6
2.2	Внутренне используемые блоки	6
3.	Добавление библиотеки	7
3.1	Интеграция в Siemens SIMATIC Manager	8
3.2	Интеграция в Siemens TIA Portal	9
4.	Обмен данными с использованием протоколов Modbus	10
4.1	TCP.....	10
4.1.1	FB 70 - TCP_MB_CLIENT – Клиент Modbus/TCP.....	10
4.1.2	FB 71 - TCP_MB_SERVER – Сервер Modbus/TCP.....	13
4.2	RTU.....	17
4.2.1	FB 72 - RTU_MB_MASTER – Ведущее устройство Modbus RTU.....	17
4.2.2	FB 73 - RTU_MB_SLAVE – Ведомое устройство Modbus RTU.....	20
4.3	Коды FKT (коды функций Modbus).....	24

1 Общие сведения

1.1 Авторские права VIPA GmbH

Все права защищены

Этот документ содержит информацию, которая является собственностью VIPA и не может разглашаться или использоваться без соответствующего разрешения или соглашения.

Этот материал защищен законами об авторских правах. Он не может быть воспроизведен, распространен или изменен каким-либо образом любым лицом (внутренним или внешним по отношению к VIPA) за исключением соответствующих действующих соглашений, контрактов или лицензий, без письменного согласия VIPA и владельца данного материала.

Для получения разрешения на воспроизведение или распространение, пожалуйста, обращайтесь: VIPA, Gesellschaft für Visualisierung und Prozessautomatisierung mbH Ohmstraße 4, D-91074 Herzogenaurach, Germany

Телефон: +49 9132 744-0

Факс: +49 9132 744-1864

Email: info@vipa.de

<http://www.vipa.com>



Было сделано всё возможное, чтобы информация, содержащаяся в данном документе, была полной и точной на момент публикации. Тем не менее, авторы оставляют за собой право вносить в неё изменения.

Этот пользовательский документ описывает все аппаратные компоненты и функции, существующие в настоящее время. Здесь может быть приведено описание устройств, отсутствующих у пользователя. Точный комплект поставки

Декларация соответствия нормам CE

Настоящим VIPA GmbH заявляет, что продукты и системы соответствуют основным требованиям директив и стандартов Европейского Союза. Соответствие подтверждается знаком CE на изделии.

Информация о соответствии

Для получения дополнительной информации относительно маркировки CE и Декларации соответствия (DoC), пожалуйста, свяжитесь с Вашим региональным представительством VIPA.

Товарные знаки

VIPA, SLIO, System 100V, System 200V, System 300V, System 300S, System 400V, System 500S и Commander Compact являются зарегистрированными торговыми знаками VIPA Gesellschaft für Visualisierung und Prozessautomatisierung mbH.

SPEED7 является зарегистрированным товарным знаком profichip GmbH.

SIMATIC, STEP, SINEC, TIA Portal, S7-300, S7-400 и S7-1500 являются зарегистрированными торговыми знаками Siemens AG.

Microsoft и Windows являются зарегистрированными торговыми знаками Microsoft Inc., USA.

Portable Document Format (PDF) и Postscript являются зарегистрированными торговыми знаками Adobe Systems, Inc.

Все другие товарные знаки, логотипы, услуги и торговые марки, указанные здесь, являются собственностью соответствующих компаний.

Информационная поддержка по продукции

Если Вы хотите сообщить об ошибках или задать вопросы относительно содержимого этого документа, свяжитесь с Вашим региональным представительством VIPA. Если Вы не можете найти местный центр обслуживания клиентов, свяжитесь с VIPA по следующему адресу:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Факс: +49 9132 744-1204

Email: documentation@vipa.de

Техническая поддержка

Если у Вас возникли проблемы при использовании продукции или есть вопросы по ней, то свяжитесь с Вашим региональным представительством VIPA. Если Вы не можете найти местный центр обслуживания клиентов, свяжитесь с VIPA по следующему адресу:

VIPA GmbH, Ohmstraße 4, 91074 Herzogenaurach, Germany

Тел.: +49 9132 744-1150 (горячая линия)

Email: support@vipa.de

1.2 Информация о руководстве

Цель и содержание

В этом руководстве приведено описание библиотеки функциональных блоков 'Modbus Communication' компании VIPA:

- Оно содержит описание структуры, процесса проектирования и применения в различных системах разработки.
- Руководство предназначено для пользователей, имеющих базовые знания по технологиям промышленной автоматизации.
- Руководство доступно в электронном формате в виде файла PDF. Для его просмотра требуется Adobe Acrobat Reader.
- Руководство разделено на главы. Каждая глава содержит детальное описание определенной темы.
- Для удобства пользования в руководстве имеются:
 - Полное содержание документа в самом его начале
 - Ссылки с номерами страниц

Предупреждающие знаки и сообщения

Важная информация в тексте выделяется следующими знаками и сообщениями:

**ОПАСНОСТЬ!**

Прямая или потенциальная опасность. Возможно получение травмы персоналом.

**ВНИМАНИЕ!**

Возможен материальный ущерб, если опасная ситуация будет проигнорирована.



Дополнительная информация и полезные советы.

2 Важные замечания

2.1 Общая информация



Ниже вы найдете важную информацию, которая всегда должна учитываться при использовании блоков.

2.2 Внутренне используемые блоки



ВНИМАНИЕ!

Следующие блоки используются внутренне и не должны быть перезаписаны! Прямой вызов внутреннего блока приводит к ошибкам в соответствующем экземплярном DB! Всегда используйте соответствующую функцию для вызова.

FC/SFC	Обозначение	Описание
FC/SFC 192	CP_S_R	используется внутренне для FB 7 и FB 8
FC/SFC 196	AG_CNTRL	используется внутренне для FC 10
FC/SFC 200	AG_GET	используется внутренне для FB/SFB 14
FC/SFC 201	AG_PUT	используется внутренне для FB/SFB 15
FC/SFC 202	AG_BSEND	используется внутренне для FB/SFB 12
FC/SFC 203	AG_BRCV	используется внутренне для FB/SFB 13
FC/SFC 204	IP_CONF	используется внутренне для FB 55 IP_CONF
FC/SFC 205	AG_SEND	используется внутренне для FC 5 AG_SEND
FC/SFC 206	AG_RECV	используется внутренне для FC 6 AG_RECV
FC/SFC 253	IBS_ACCESS	используется внутренне для ведущих устройств INTERBUS на шине SPEED Bus
SFB 238	EC_RWOD	используется внутренне для обмена данными через EtherCAT
SFB 239	FUNC	используется внутренне для FB 240, FB 241

3 Добавление библиотеки

Библиотека функциональных блоков 'Modbus Communication'

Библиотеку блоков можно загрузить с сервисной зоны 'Service/Support' веб-сайта www.vipa.com из раздела 'Downloads → VIPA Lib' как 'Block library Modbus Communication - SW90AS0MA'. Библиотека доступна в виде архивных zip-файлов. Для использования блоков их предварительно необходимо импортировать в свой проект.



Всегда используйте руководство, относящееся к используемой библиотеке. При этом номер версии руководства может отличаться от версии библиотеки и входящих в нее файлов.

Доступны следующие библиотеки функциональных блоков

Файл	Описание
Modbus_S7_V0004.zip	<ul style="list-style-type: none"> ■ Функциональная библиотека для Siemens SIMATIC Manager. ■ Для использования в процессорных модулях VIPA или ЦПУ S7-300 Siemens.
Modbus_TIA_V14_V0005.zip	<ul style="list-style-type: none"> ■ Функциональная библиотека для Siemens TIA Portal V14. ■ Для использования в процессорных модулях VIPA или ЦПУ S7-300 Siemens.
Modbus_TIA_V11_V13_V0003.zip	<ul style="list-style-type: none"> ■ Функциональная библиотека для Siemens TIA Portal V11 и V13. ■ Для использования в процессорных модулях VIPA или ЦПУ S7-300 Siemens.

3.1 Интеграция в Siemens SIMATIC Manager

Обзор	<p>Для интеграции в Siemens SIMATIC Manager необходимо выполнить следующие действия:</p> <ol style="list-style-type: none">1. ➤ Загрузить zip-файл2. ➤ Извлечь библиотеку из архива3. ➤ Открыть библиотеку и перенести блоки в проект
Загрузка zip-файла	<p>➤ Перейдите на веб-страницу к требуемому zip-файлу, загрузите его и сохраните в рабочей папке.</p>
Извлечение библиотеки	<ol style="list-style-type: none">1. ➤ Запустите SIMATIC Manager компании Siemens со своим проектом.2. ➤ Откройте диалоговое окно для выбора ZIP-файла с помощью <i>'File → Retrieve'</i>.3. ➤ Выберите нужный ZIP-файл и кликните по [Open].4. ➤ Укажите папку, в которую будут сохранены блоки.5. ➤ Запустите процесс распаковки с помощью [OK].
Открытие библиотеки и перенос блоков в проект	<ol style="list-style-type: none">1. ➤ Откройте библиотеку после её распаковки.2. ➤ Откройте свой проект и скопируйте нужные блоки из библиотеки в раздел "Blocks" активного проекта. <p>⇒ Теперь специализированные функциональные блоки VIPA готовы для использования в пользовательской программе проекта.</p>



Использование FC вместо SFC поддерживается модулями ЦПУ VIPA с прошивкой 3.6.0 и выше.

3.2 Интеграция в Siemens TIA Portal

Обзор

Для интеграции в Siemens TIA Portal необходимо выполнить следующие действия:

1. ➤ Загрузить zip-файл
2. ➤ Распаковать zip-файл
3. ➤ Извлечь библиотеку из архива
4. ➤ Открыть библиотеку и перенести блоки в проект

Загрузка zip-файла

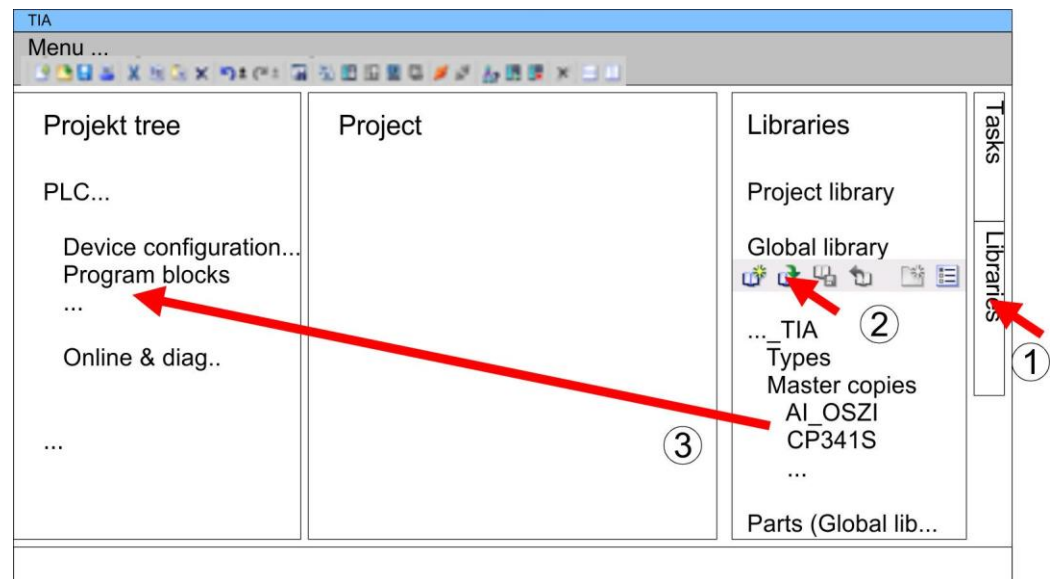
1. ➤ Перейдите на веб-страницу к требуемому zip-файлу, соответствующему используемой системе разработки.
2. ➤ Загрузите его и сохраните в рабочей папке.

Распаковка zip-файла

- Распакуйте ZIP-файл с помощью программы распаковки в рабочий каталог Siemens TIA Portal.

Открытие библиотеки и перенос блоков в проект

1. ➤ Запустите Siemens TIA Portal со своим проектом.
2. ➤ Перейдите в *Project view*.
3. ➤ Выберите вкладку "Libraries" из вертикального меню справа.
4. ➤ Кликните по "Global library".
5. ➤ Кликните по "Open global library".
6. ➤ Перейдите в рабочий каталог и загрузите файл ..._TIA.al1x.



7. ➤ Скопируйте нужные блоки из библиотеки в раздел "Program blocks" менеджера проектов *Project tree* активного проекта. Теперь специализированные функциональные блоки VIPA готовы для использования в пользовательской программе проекта.

4 Обмен данными с использованием протоколов Modbus

4.1 TCP

4.1.1 FB 70 - TCP_MB_CLIENT – Клиент Modbus/TCP

4.1.1.1 Описание

Этот функциональный блок позволяет использовать порт Ethernet в качестве клиента Modbus/TCP.

Параметры вызова

Наименование	Тип переменной	Тип данных	Описание
REQ	ВХОД	BOOL (Двоичное значение)	Запуск работы блока по переходу 0-1.
ID	ВХОД	WORD (16-разрядное значение)	ID из TCON.
MB_FUNCTION	ВХОД	BYTE (Байт)	Modbus: Код функции.
MB_DATA_ADDR	ВХОД	WORD (16-разрядное значение)	Modbus: Начальный адрес или код подфункции.
MB_DATA_LEN	ВХОД	INT (Целое)	Modbus: Количество регистров/бит
MB_DATA_PTR	ВХОД	ANY (любой)	Modbus: Буфер данных (допустима только область флагов или блок данных с типом данных BYTE) при использовании функций с кодами 03h, 06h и 10h.
DONE *	ВЫХОД	BOOL (Двоичное значение)	Работа блока завершена без ошибок.
BUSY	ВЫХОД	BOOL (Двоичное значение)	Задание выполняется.
ERROR *	ВЫХОД	BOOL (Двоичное значение)	Работа блока завершена с ошибкой - Параметр STATUS содержит информацию об ошибке.
STATUS *	ВЫХОД	WORD (16-разрядное значение)	Расширенная информация о состоянии и об ошибках.

*) Параметр сохраняет свое состояние до следующего вызова FB

Параметры экземплярного DB

Наименование	Тип переменной	Тип данных	Описание
PROTOCOL_TIMEOUT	STAT	INT (Целое)	Время блокировки до того, как пользователь может отменить активное задание. Значение по умолчанию: 3 с
RCV_TIMEOUT	STAT	INT (Целое)	Контрольное время для задания. Значение по умолчанию: 2 с
MB_TRANS_ID	STAT	WORD (16-разрядное значение)	Modbus: Начальное значение для идентификатора транзакции. Значение по умолчанию: 1
MB_UNIT_ID	STAT	BYTE (Байт)	Modbus: Идентификация устройства. Значение по умолчанию: 255

Обратите внимание на следующее:

- *Параметры вызова* должны быть заданы при вызове блока. Кроме *параметров вызова* имеются также параметры, содержащиеся в экземплярном DB.
- Предварительно коммуникационное соединение должно быть инициализировано с помощью FB 65 (TCON).
- Для использования блока требуются FB 63 (TSEND) и FB 64 (TRCV).
- В процессе исполнения задания экземплярный DB блокируется для других клиентов.

- В процессе исполнения задания изменение состояния входных параметров не воспринимается.
- Исполнение задания завершается или отменяется при одном из следующих условий:
 - DONE = 1 (исполнение без ошибки)
 - ERROR = 1 (исполнение с ошибкой)
 - Превышение интервала RCV_TIMEOUT
 - REQ = FALSE после истечения времени PROTOCOL_TIMEOUT
- Если REQ сбрасывается до того, как установлен выход DONE или ERROR или истекло время PROTOCOL_TIMEOUT, на выходе STATUS будет установлено значение 8200h. При этом текущее задание будет по-прежнему исполняться.

Отображение состояния и ошибок

Функциональный блок выводит на выход STATUS информацию о состоянии и об ошибках.

STATUS	DONE	BUSY	ERROR	Описание
0000h	1	0	0	Задание выполнено без ошибки.
7000h	0	0	0	Соединение не установлено или ошибка связи (TCON).
7004h	0	0	0	Соединение установлено и контролируется. Активное задание отсутствует.
7005h	0	1	0	Данные передаются.
7006h	0	1	0	Данные принимаются.
8210h	0	0	1	Аппаратное обеспечение несовместимо с библиотекой функциональных блоков Modbus RTU/TCP.
8380h	0	0	1	Принятое сообщение Modbus имеет некорректный формат или недопустимую длину.
8381h	0	0	1	Сервер возвращает <i>Exception code 01h</i> .
8382h	0	0	1	Сервер возвращает <i>Exception code 03h</i> или неправильный стартовый адрес.
8383h	0	0	1	Сервер возвращает <i>Exception code 02h</i> .
8384h	0	0	1	Сервер возвращает <i>Exception code 04h</i> .
8386h	0	0	1	Сервер возвращает неверный код функции.
8387h	0	0	1	Идентификатор соединения (TCON) не соответствует экземпляру блока или сервер возвращает неверный идентификатор протокола.
8388h	0	0	1	Сервер возвращает неверное значение или неверное количество.
80C8h	0	0	1	Отсутствие ответа сервера в течение заданного интервала (RCV_TIMEOUT).
8188h	0	0	1	Недопустимое значение MB_FUNCTION.
8189h	0	0	1	Недопустимое значение MB_DATA_ADDR.
818Ah	0	0	1	Недопустимое значение MB_DATA_LEN.
818Bh	0	0	1	Недопустимое значение MB_DATA_PTR.
818Ch	0	0	1	Недопустимое значение BLOCKED_PROC_TIMEOUT или RCV_TIMEOUT.
818Dh	0	0	1	Сервер возвращает неверный идентификатор транзакции.
8200h	0	0	1	Другой запрос Modbus через порт обрабатывается в данный момент (PROTOCOL_TIMEOUT).

TCP > FB 70 - TCP_MB_CLIENT - Клиент Modbus/TCP

4.1.1.2 Пример

Задача

С помощью функции 03h с сервера Modbus/TCP должны быть считаны 100 регистров, начиная с адреса 2000, и сохраняться в области памяти флагов с начальным адресом MB200. Ошибки должны быть сохранены.

OB1

```

CALL FB 65 , DB65
  REQ      :=M100.0
  ID       :=W#16#1
  DONE     :=M100.1
  BUSY     :=
  ERROR    :=M100.2
  STATUS   :=MW102
  CONNECT:=P#DB255.DBX 0.0 BYTE 64

UN      M      100.2
SPB     ERR1
L       MW     102
T       MW     104
ERR1:  NOP    0
U       M      100.1
R       M      100.0

CALL FB 70 , DB70
  REQ      :=M101.0
  ID       :=W#16#1
  MB_FUNCTION :=B#16#3
  MB_DATA_ADDR:=W#16#7D0
  MB_DATA_LEN :=100
  MB_DATA_PTR :=P#M 200.0 BYTE 200
  DONE     :=M101.1
  BUSY     :=
  ERROR    :=M101.2
  STATUS   :=MW106

UN      M      101.2
SPB     ERR2
L       MW     106
T       MW     108
ERR2:  NOP    0
U       M      101.1
R       M      101.0

```

OB1 - Описание

1. ➤ Вызов FB 65 (TCON) для установления соединения с партнёрской станцией.
2. ➤ Вызов блока обработки данных клиента Modbus/TCP с корректными параметрами.
3. ➤ Соединение с партнёрской станцией отсутствует, ячейка памяти MW102 содержит код ошибки 7000h.
4. ➤ Установка в ЦПУ для M100.0 значения TRUE.
⇒ Если M100.0 автоматически сбрасывается, то это означает, что соединение установлено и ячейка памяти MW108 в этом случае содержит код 7004h.
5. ➤ Установка в ЦПУ для M101.0 значения TRUE.
⇒ Посылается запрос Modbus и ждётся ответ.

Если M101.0 сбрасывается автоматически, задание исполнено без ошибок, а считанные данные сохранены в ЦПУ в области памяти флагов, начиная с байта 200. MW108 содержит код 7004h, что сигнализирует о готовности к выполнению нового запроса.

Если M101.0 не сбрасывается автоматически, а MW108 возвращает значение, отличное от 0, то это означает наличие ошибки обмена. Причину ошибки можно определить с помощью кода в MW108 (например, MW108 = 8382h, если начальный адрес 2000 отсутствует на сервере). MW108 содержит код 7004h, что сигнализирует о готовности к выполнению нового запроса.

4.1.2 TCP > FB 71 - TCP_MB_SERVER - Сервер Modbus/TCP

4.1.2.1 Описание

Этот функциональный блок позволяет использовать порт Ethernet в качестве сервера Modbus/TCP.

Параметры вызова

Наименование	Тип переменной	Тип данных	Описание
ENABLE	ВХОД	BOOL (Двоичное значение)	Активация/деактивация сервера Modbus.
MB_DATA_PTR	ВХОД	ANY (любой)	Modbus: Буфер данных (допустима только область флагов или блок данных с типом данных BYTE) при использовании функций с кодами 03h, 06h и 10h.
ID	ВХОД	WORD (16-разрядное значение)	ID из TCON.
NDR*	ВЫХОД	BOOL (Двоичное значение)	Новые данные были записаны клиентом Modbus.
DR*	ВЫХОД	BOOL (Двоичное значение)	Данные были считаны клиентом Modbus.
ERROR*	ВЫХОД	BOOL (Двоичное значение)	Работа блока завершена с ошибкой - Параметр STATUS содержит информацию об ошибке.
STATUS*	ВЫХОД	WORD (16-разрядное значение)	Расширенная информация о состоянии и об ошибках.

*) Параметр сохраняет свое состояние до следующего вызова FB

Параметры экземплярного DB

Наименование	Тип переменной	Тип данных	Описание
REQUEST_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого полученного сообщения.
MESSAGE_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого корректного запроса Modbus.
XMT_RCV_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого полученного сообщения, которое не содержит корректного запроса Modbus.
EXCEPTION_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого отрицательно подтвержденного запроса Modbus.
SUCCESS_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого положительно подтвержденного запроса Modbus.
FC1_ADDR_OUTPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Начальный регистр для Q0.0 Значение по умолчанию: 0
FC1_ADDR_OUTPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Конечный регистр для Qx.y Значение по умолчанию: 19999
FC1_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Начальный регистр для M0.0 Значение по умолчанию: 20000
FC1_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Конечный регистр для Mx.y Значение по умолчанию: 39999
FC2_ADDR_INPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Начальный регистр для I0.0 Значение по умолчанию: 0

TCP > FB 71 - TCP_MB_SERVER - Сервер Modbus/TCP

Наименование	Тип переменной	Тип данных	Описание
FC2_ADDR_INPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Конечный регистр для Ix.y Значение по умолчанию: 19999
FC2_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Начальный регистр для M0.0 Значение по умолчанию: 20000
FC2_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Конечный регистр для Mx.y Значение по умолчанию: 39999
FC4_ADDR_INPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Начальный регистр для IW0 Значение по умолчанию: 0
FC4_ADDR_INPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Конечный регистр для IWx Значение по умолчанию: 19999
FC4_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Начальный регистр для MW0 Значение по умолчанию: 20000
FC4_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Конечный регистр для MWx Значение по умолчанию: 39999
FC5_ADDR_OUTPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Начальный регистр для Q0.0 Значение по умолчанию: 0
FC5_ADDR_OUTPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Конечный регистр для Qx.y Значение по умолчанию: 19999
FC5_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Начальный регистр для M0.0 Значение по умолчанию: 20000
FC5_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Конечный регистр для Mx.y Значение по умолчанию: 39999
FC15_ADDR_OUTPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Начальный регистр для Q0.0 Значение по умолчанию: 0
FC15_ADDR_OUTPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Конечный регистр для Qx.y Значение по умолчанию: 19999
FC15_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Начальный регистр для Q0.0 Значение по умолчанию: 20000
FC15_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Конечный регистр для Qx.y Значение по умолчанию: 39999

Обратите внимание на следующее:

- *Параметры вызова* должны быть заданы при вызове блока. Кроме *параметров вызова* имеются также параметры, содержащиеся в экземплярном DB.
- Предварительно коммуникационное соединение должно быть инициализировано с помощью FB 65 (TCN).
- Для использования блока требуются FB 63 (TSEND) и FB 64 (TRCV).
- Адреса Modbus INPUT/OUTPUT для конкретной используемой функции Modbus должны быть расположены перед адресом MEMORY Modbus и, соответственно, всегда меньше по значению.

- Для одного кода функции адрес Modbus не может быть определен более одного раза - даже со значением 0!
- Сервер может одновременно обрабатывать только одно задание. Новые запросы Modbus во время обработки задания игнорируются, ответ на них не формируется.

Отображение состояния и ошибок

Функциональный блок выводит на выход STATUS информацию о состоянии и об ошибках.

STATUS	NDR	DR	ERROR	Описание
0000h	0 или 1*		0	Задание выполнено без ошибки.
7000h	0	0	0	Соединение не установлено или ошибка связи (TCON).
7005h	0	0	0	Данные передаются.
7006h	0	0	0	Данные принимаются.
8210h	0	0	1	Аппаратное обеспечение несовместимо с библиотекой функциональных блоков Modbus RTU/TCP.
8380h	0	0	1	Принятое сообщение Modbus имеет некорректный формат или получено меньшее количество байт данных.
8381h	0	0	1	<i>Exception code 01h</i> , код функции не поддерживается.
8382h	0	0	1	<i>Exception code 03h</i> , длина данных или значение данных недействительны.
8383h	0	0	1	<i>Exception code 02h</i> , недопустимый начальный адрес или диапазон адресов.
8384h	0	0	1	<i>Exception code 04h</i> , ошибка длины диапазона при доступе к входам, выходам или флагам.
8387h	0	0	1	Идентификатор соединения (TCON) не соответствует экземпляру блока или клиент возвращает неверный идентификатор протокола.
8187h	0	0	1	Недопустимое значение MB_DATA_PTR.

*) Исполненный без ошибок запрос Modbus с кодами функций 05h, 06h, 0Fh или 10h возвращает NDR=1 и DR=0.

Исполненный без ошибок запрос Modbus с кодами функций 01h, 02h, 03h или 04h возвращает NDR=0 и DR=1.

TCP > FB 71 - TCP_MB_SERVER - Сервер Modbus/TCP

4.1.2.2 Пример

Задача

ЦПУ помещает 100 байт данных в область флагов клиента Modbus с начальным адресом MB200, используя для этого регистры Modbus 0...49. Данные могут быть считаны из клиента Modbus с использованием *функции 03h* и записаны с использованием *функций 06h и 10h*. Выход Q1.0 модуля ЦПУ должен управляться клиентом Modbus, используя для этого *функцию 05h* и начальный адрес 5008. Значения ошибок должны быть сохранены.

OB1

```
CALL FB 65 , DB65
    REQ      :=M100.0
    ID       :=W#16#1
    DONE     :=M100.1
    BUSY     :=
    ERROR    :=M100.2
    STATUS   :=MW102
    CONNECT:=P#DB255.DBX 0.0 BYTE 64

    UN      M    100.2
    SPB     ERR1

    L       MW   102
    T       MW   104

ERR1: NOP 0
    U      M    100.1
    R      M    100.0

    L      5000
    T     DB71.DBW 52

CALL FB 71 , DB71
    ENABLE     :=M101.0
    MB_DATA_PTR:=P#M 200.0 BYTE 100
    ID         :=W#16#1
    NDR        :=M101.1
    DR         :=M101.2
    ERROR      :=M101.3
    STATUS     :=MW106

    UN      M    101.3
    SPB     ERR2
    L       MW   106
    T       MW   108

ERR2: NOP 0
```

OB1 - Описание

1. ➤ Вызов FB 65 (TCON) для установления соединения с партнёрской станцией.
2. ➤ Вызов блока обработки данных сервера Modbus/TCP с корректными параметрами.
3. ➤ Отсутствие соединения с партнёрской станцией, ячейка памяти MW102 содержит код ошибки 7000h.
4. ➤ Установка в ЦПУ для M100.0 значения TRUE.
⇒ Если M100.0 автоматически сбрасывается, то это означает, что соединение с партнёрской станцией установлено и ячейка памяти MW108 в этом случае содержит код 7006h.
5. ➤ Начальный регистр Modbus для *функции 05h*, используемой для записи данных в образ процесса, в приведенном примере может изменяться через параметр FC5_ADDR_OUTPUT_START (слово 52 в экземплярном блоке данных).
6. ➤ Установка в ЦПУ для M101.0 значения TRUE.
⇒ Сервер Modbus теперь запущен.
7. ➤ Клиент Modbus отправляет запрос с *кодом функции 03h*, начальным адресом 10 и количеством 30.

⇒ Сервер отвечает 60 байтами данных, расположенными в памяти ЦПУ, начиная с адреса MB220. DR устанавливается на время одного цикла ЦПУ и, следовательно, значение M101.2 равно "1".

8. Клиент Modbus направляет запрос с *кодом функции 05h*, начальным адресом 5008 и значением FF00h.

⇒ Сервер подтверждает запрос и записывает "1" на выход Q1.0. NDR устанавливается на время одного цикла ЦПУ и, следовательно, значение M101.1 равно "1".

9. Клиент Modbus направляет запрос с *кодом функции 03h*, начальным адресом 50 (не существует!) и количеством 1.

⇒ Сервер отвечает с *Exception code 02h* и устанавливает на выходах ERROR/STATUS соответствующие значения на время одного цикла ЦПУ. MW108 содержит значение 8383h.

4.2 RTU

4.2.1 RTU > FB 72 - RTU_MB_MASTER - Ведущее устройство Modbus RTU

4.2.1.1 Описание

Этот функциональный блок обеспечивает работу внутреннего порта процессорного модуля VIPA SPEED7 с последовательным интерфейсом RS-485, а также модуля VIPA CP 040 серии SLIO в режиме ведущего устройства Modbus RTU.

Параметры вызова

Наименование	Тип переменной	Тип данных	Описание
REQ	ВХОД	BOOL (Двоичное значение)	Запуск работы блока по переходу 0-1.
HARDWARE	ВХОД	BYTE (Байт)	1 = Модуль SLIO CP 040 / 2 = ЦПУ VIPA SPEED7
LADDR	ВХОД	INT (Целое)	Логический адрес модуля SLIO CP 040 (параметр игнорируется для ЦПУ VIPA SPEED7).
MB_UNIT_ID	ВХОД	BYTE (Байт)	Modbus: Идентификатор устройства = Адрес ведомого устройства (0 ... 247).
MB_FUNCTION	ВХОД	BYTE (Байт)	Modbus: Код функции.
MB_DATA_ADDR	ВХОД	WORD (16-разрядное значение)	Modbus: Начальный адрес или код подфункции.
MB_DATA_LEN	ВХОД	INT (Целое)	Modbus: Количество регистров/бит
MB_DATA_PTR	ВХОД	ANY (любой)	Modbus: Буфер данных (допустима только область флагов или блок данных с типом данных BYTE) при использовании функций с кодами 03h, 06h и 10h.
DONE*	ВЫХОД	BOOL (Двоичное значение)	Работа блока завершена без ошибок.
BUSY	ВЫХОД	BOOL (Двоичное значение)	Задание выполняется.
ERROR*	ВЫХОД	BOOL (Двоичное значение)	Работа блока завершена с ошибкой - Параметр STATUS содержит информацию об ошибке.
STATUS*	ВЫХОД	WORD (16-разрядное значение)	Расширенная информация о состоянии и об ошибках.

*) Параметр сохраняет свое состояние до следующего вызова FB

Параметры экземплярного DB

Наименование	Тип переменной	Тип данных	Описание
INIT	STAT	BOOL (Двоичное значение)	По переходу 0-1 выполняется синхронный сброс модуля SLIO CP 040. После успешного выполнения сброса бит сбрасывается автоматически.

RTU > FB 72 - RTU_MB_MASTER - Ведущее устройство Modbus RTU

Обратите внимание на следующее:

- *Параметры вызова* должны быть заданы при вызове блока. Кроме *параметров вызова* имеются также параметры, содержащиеся в экземплярном DB.
- Используемый интерфейс необходимо предварительно настроить:
 - Модуль SLIO CP 040: в конфигураторе оборудования задается как устройство "Modbus master RTU" с размером адресного пространства ввода-вывода 60 байт.
 - Внутренний порт с последовательным интерфейсом RS-485 процессорного модуля VIPA: конфигурирование с помощью SFC 216 (SER_CFG) с протоколом "Modbus master RTU".
- Функциональные блоки FB 60 SEND и FB 61 RECEIVE (или FB 65 SEND_RECV) абсолютно необходимы для использования блока, даже если используется внутренний последовательный интерфейс RS-485 модуля ЦПУ VIPA.
- В процессе исполнения задания изменение состояния входных параметров не воспринимается.
- Запросы широковещательной передачи с помощью MB_UNIT_ID = 0 возможны только для функций записи.
- Исполнение задания завершается или отменяется при одном из следующих условий:
 - DONE = 1 (завершение без ошибки)
 - ERROR = 1 (завершение с ошибкой)
 - Истечение времени ожидания (параметризация интерфейса)
- Если REQ сбрасывается до того, как установлен выход DONE или ERROR, на выходе STATUS будет установлено значение 8200h. При этом текущее задание будет по-прежнему исполняться.

Отображение состояния и ошибок

Функциональный блок выводит на выход STATUS информацию о состоянии и об ошибках.

STATUS	DONE	BUSY	ERROR	Описание
0000h	1	0	0	Задание выполнено без ошибки.
7000h	0	0	0	Соединение не установлено или ошибка связи.
7004h	0	0	0	Соединение установлено и контролируется. Активное задание отсутствует.
7005h	0	1	0	Данные передаются.
7006h	0	1	0	Данные принимаются.
8210h	0	0	1	Аппаратное обеспечение несовместимо с библиотекой функциональных блоков Modbus RTU/TCP.
8381h	0	0	1	Сервер возвращает <i>Exception code 01h</i> .
8382h	0	0	1	Сервер возвращает <i>Exception code 03h</i> или неправильный стартовый адрес.
8383h	0	0	1	Сервер возвращает <i>Exception code 02h</i> .
8384h	0	0	1	Сервер возвращает <i>Exception code 04h</i> .
8386h	0	0	1	Сервер возвращает неверный код функции.
8388h	0	0	1	Сервер возвращает неверное значение или неверное количество.
80C8h	0	0	1	Отсутствие ответа сервера в течение заданного интервала (значение параметрируется через интерфейс).
8188h	0	0	1	Недопустимое значение MB_FUNCTION.
8189h	0	0	1	Недопустимое значение MB_DATA_ADDR.
818Ah	0	0	1	Недопустимое значение MB_DATA_LEN.
818Bh	0	0	1	Недопустимое значение MB_DATA_PTR.
8201h	0	0	1	Недопустимое оборудование.
8202h	0	0	1	Недопустимое значение MB_UNIT_ID.
8200h	0	0	1	Другой запрос Modbus в настоящее время обслуживается портом.

4.2.1.2 Пример

Задача

С помощью функции 03h из ведомого устройства Modbus RTU должны быть считаны 100 регистров, начиная с адреса 99, и сохранены в области памяти флагов с начальным адресом MB200. Значения ошибок должны быть сохранены. Ведущее устройство Modbus RTU реализуется на основе внутреннего порта с последовательным интерфейсом RS-485 процессорного модуля VIPA.

OB100

```
CALL SFC 216
  Protocol :=B#16#5
  Parameter :=DB10
  Baudrate:=B#16#9
  CharLen:=B#16#3
  Parity:=B#16#2
  StopBits:=B#16#1
  FlowControl:=B#16#1
  RetVal:=MW100
```

OB100 - Описание

1. ➤ Вызов SFC 216 (SER_CFG) для настройки внутреннего последовательного интерфейса ЦПУ VIPA.
2. ➤ Протокол: "Modbus Master RTU", 9600 бод, 8 бит данных, 1 стоповый бит, контроль по чётности, отсутствие контроля потока данных.
3. ➤ Блок данных DB10 содержит переменную типа WORD со значением таймаута для Modbus (значение в мс).

OB1

```
CALL FB 72 , DB72
  REQ      :=M101.0
  HARDWARE :=B#16#2
  LADDR    :=
  MB_UNIT_ID :=B#16#63
  MB_FUNCTION :=B#16#3
  MB_DATA_ADDR:=W#16#7D0
  MB_DATA_LEN :=100
  MB_DATA_PTR :=P#M 200.0 BYTE 200
  DONE      :=M101.1
  BUSY      :=
  ERROR     :=M101.2
  STATUS    :=MW102

  UN      M      101.2
  SPB     ERR1
  L       MW     102
  T       MW     104
ERR1: NOP 0
  U      M      101.1
  R      M      101.0
```

OB1 - Описание

1. ➤ Вызов блока обработки данных ведущего устройства Modbus RTU с корректными параметрами.
2. ➤ Если интерфейс был корректно инициализирован в OB 100, ведущее устройство готово к работе и MW102 содержит 7004h.

3. Установка в ЦПУ для M101.0 значения TRUE.

⇒ Отправка запроса Modbus и ожидание ответа.

Если M101.0 сбрасывается автоматически, задание исполнено без ошибок, а считанные данные сохранены в ЦПУ в области памяти флагов, начиная с байта 200. MW104 содержит код 7004h, что сигнализирует о готовности к выполнению нового запроса.

Если M101.0 не сбрасывается автоматически, а MW104 возвращает значение, отличное от 0, то это означает наличие ошибки обмена. Причину ошибки можно определить с помощью кода в MW104 (например, MW104 = 8382h, если начальный адрес 2000 отсутствует в ведомом устройстве). MW102 содержит код 7004h, что сигнализирует о готовности к выполнению нового запроса.

4.2.2 FB 73 - RTU_MB_SLAVE - Ведомое устройство Modbus RTU**4.2.2.1 Описание**

Этот функциональный блок обеспечивает работу внутреннего порта процессорного модуля VIPA SPEED7 с последовательным интерфейсом RS-485, а также модуля VIPA CP 040 серии SLIO в режиме ведомого устройства Modbus RTU.

Параметры вызова

Наименование	Тип переменной	Тип данных	Описание
ENABLE	ВХОД	BOOL (Двоичное значение)	Активация/деактивация сервера Modbus.
HARDWARE	ВХОД	BYTE (Байт)	1 = Модуль SLIO CP 040 / 2 = ЦПУ VIPA SPEED7
LADDR	ВХОД	INT (Целое)	Логический адрес модуля SLIO CP 040 (параметр игнорируется для ЦПУ VIPA SPEED7).
MB_UNIT_ID	ВХОД	BYTE (Байт)	Modbus: Идентификатор устройства = Собственный адрес устройства (1 ... 247).
MB_DATA_PTR	ВХОД	ANY (любой)	Modbus: Буфер данных (допустима только область флагов или блок данных с типом данных BYTE) при использовании функций с кодами 03h, 06h и 10h.
NDR*	ВЫХОД	BOOL (Двоичное значение)	Новые данные были записаны клиентом Modbus.
DR*	ВЫХОД	BOOL (Двоичное значение)	Данные были считаны клиентом Modbus.
ERROR*	ВЫХОД	BOOL (Двоичное значение)	Работа блока завершена с ошибкой - Параметр STATUS содержит информацию об ошибке.
STATUS*	ВЫХОД	WORD (16-разрядное значение)	Расширенная информация о состоянии и об ошибках.

*) Параметр сохраняет свое состояние до следующего вызова FB

Параметры экземплярного DB

Наименование	Тип переменной	Тип данных	Описание
INIT	STAT	BOOL (Двоичное значение)	По переходу 0-1 выполняется синхронный сброс модуля SLIO CP 040.
REQUEST_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого полученного сообщения.
MESSAGE_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого корректного запроса Modbus.
BROADCAST_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого корректного широковещательного запроса Modbus.

Наименование	Тип переменной	Тип данных	Описание
EXCEPTION_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого отрицательно подтвержденного запроса Modbus.
SUCCESS_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого положительно подтвержденного запроса Modbus.
BAD_CRC_COUNT	STAT	WORD (16-разрядное значение)	Счетчик для каждого корректного запроса Modbus с ошибкой CRC.
FC1_ADDR_OUTPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Начальный регистр для Q0.0. Значение по умолчанию: 0
FC1_ADDR_OUTPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Конечный регистр для Qx.y. Значение по умолчанию: 19999
FC1_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Начальный регистр для M0.0. Значение по умолчанию: 20000
FC1_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 01h. Конечный регистр для Mx.y. Значение по умолчанию: 39999
FC2_ADDR_INPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Начальный регистр для I0.0. Значение по умолчанию: 0
FC2_ADDR_INPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Конечный регистр для Ix.y. Значение по умолчанию: 19999
FC2_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Начальный регистр для M0.0. Значение по умолчанию: 20000
FC2_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 02h. Конечный регистр для Mx.y. Значение по умолчанию: 39999
FC4_ADDR_INPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Начальный регистр для IW0. Значение по умолчанию: 0
FC4_ADDR_INPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Конечный регистр для IWx. Значение по умолчанию: 19999
FC4_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Начальный регистр для MW0. Значение по умолчанию: 20000
FC4_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 04h. Конечный регистр для MW0. Значение по умолчанию: 39999
FC5_ADDR_OUTPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Начальный регистр для Q0.0. Значение по умолчанию: 0
FC5_ADDR_OUTPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Конечный регистр для Qx.y. Значение по умолчанию: 19999
FC5_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Начальный регистр для M0.0. Значение по умолчанию: 20000
FC5_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 05h. Конечный регистр для Mx.y. Значение по умолчанию: 39999

RTU > FB 73 - RTU_MB_SLAVE - Ведомое устройство Modbus RTU

Наименование	Тип переменной	Тип данных	Описание
FC15_ADDR_OUTPUT_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Начальный регистр для Q0.0. Значение по умолчанию: 0
FC15_ADDR_OUTPUT_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Конечный регистр для Qx.y. Значение по умолчанию: 19999
FC15_ADDR_MEMORY_START	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Начальный регистр для M0.0. Значение по умолчанию: 20000
FC15_ADDR_MEMORY_END	STAT	WORD (16-разрядное значение)	Код функции Modbus 0Fh. Конечный регистр для Mx.y. Значение по умолчанию: 39999

Обратите внимание на следующее:

- *Параметры вызова* должны быть заданы при вызове блока. Кроме *параметров вызова* имеются также параметры, содержащиеся в экземплярном DB.
- Используемый интерфейс необходимо предварительно настроить:
 - Модуль SLIO CP 040: в конфигураторе оборудования конфигурируется как модуль ASCII с адресным пространством ввода-вывода 60 байт.
 - Внутренний порт с последовательным интерфейсом RS-485 процессорного модуля VIPA: конфигурирование с помощью SFC 216 (SER_CFG) с протоколом "ASCII".
- Функциональные блоки FB 60 SEND и FB 61 RECEIVE (или FB 65 SEND_RECV) абсолютно необходимы для использования блока, даже если используется внутренний последовательный интерфейс RS-485 модуля ЦПУ VIPA.
- Запросы широковещательной передачи с помощью MB_UNIT_ID = 0 возможны только для функций записи.
- Адреса Modbus INPUT/OUTPUT для конкретной используемой функции Modbus должны быть расположены перед адресом MEMORY Modbus и, соответственно, всегда меньше по значению.
- Для одного кода функции адрес Modbus не может быть определен более одного раза - даже со значением 0!
- Ведомое устройство может одновременно обрабатывать только одно задание. Новые запросы Modbus во время обработки задания игнорируются, ответ на них не формируется.

Отображение состояния и ошибок

Функциональный блок выводит на выход STATUS информацию о состоянии и об ошибках.

STATUS	NDR	DR	ERROR	Описание
0000h	0 или 1*		0	Задание выполнено без ошибки.
7000h	0	0	0	Соединение не установлено или ошибка связи.
7005h	0	0	0	Данные передаются.
7006h	0	0	0	Данные принимаются.
8210h	0	0	1	Аппаратное обеспечение несовместимо с библиотекой функциональных блоков Modbus RTU/TCP.
8380h	0	0	1	Ошибка CRC
8381h	0	0	1	<i>Exception code 01h</i> , код функции не поддерживается.
8382h	0	0	1	<i>Exception code 03h</i> , длина данных или значение данных недействительны.
8383h	0	0	1	<i>Exception code 02h</i> , недопустимый начальный адрес или диапазон адресов.
8384h	0	0	1	<i>Exception code 04h</i> , ошибка длины диапазона при доступе ко входам, выходам или флагам.
8187h	0	0	1	Недопустимое значение MB_DATA_PTR.

STATUS	NDR	DR	ERROR	Описание
8201h	0	0	1	Недопустимое оборудование.
8202h	0	0	1	Недопустимое значение MB_UNIT_ID.
8203h	0	0	1	

*) Исполненный без ошибок запрос Modbus с кодами функций 05h, 06h, 0Fh или 10h возвращает NDR=1 и DR=0.

Исполненный без ошибок запрос Modbus с кодами функций 01h, 02h, 03h или 04h возвращает NDR=0 и DR=1.

4.2.2.2 Пример

Задача

ЦПУ помещает 100 байт данных в область флагов клиента Modbus с начальным адресом MB200, используя для этого регистры Modbus 0...49. Данные могут быть считаны ведущим устройством Modbus с использованием функции 03h и записаны с использованием функций 06h и 10h. Выход Q1.0 модуля ЦПУ должен управляться ведущим устройством Modbus, используя для этого функцию 05h и начальный адрес 5008. Значения ошибок должны быть сохранены. Ведомое устройство Modbus RTU с адресом 99 реализуется на основе внутреннего порта с последовательным интерфейсом RS-485 процессорного модуля VIPA.

OB100

```
CALL SFC 216
  Protocol :=B#16#1
  Parameter :=DB10
  Baudrate:=B#16#9
  CharLen:=B#16#3
  Parity:=B#16#2
  StopBits:=B#16#1
  FlowControl:=B#16#1
  RetVal:=MW100
```

OB100 - Описание

1. Вызов SFC 216 (SER_CFG) для настройки внутреннего последовательного интерфейса ЦПУ VIPA.
2. Протокол: «ASCII», 9600 бод, 8 бит данных, 1 стоповый бит, контроль по чётности, отсутствие контроля потока данных.
3. Блок данных DB10 содержит переменную типа WORD, которая должна быть передана как "Dummy".

OB1

```
L      5000
T      DB73.DBW  58

CALL FB 73 , DB73
  ENABLE      :=M101.0
  HARDWARE    :=B#16#2
  LADDR       :=
  MB_UNIT_ID  :=B#16#63
  MB_DATA_PTR:=P#M 200.0 BYTE 100
  NDR         :=M101.1
  DR          :=M101.2
  ERROR       :=M101.3
  STATUS      :=MW102

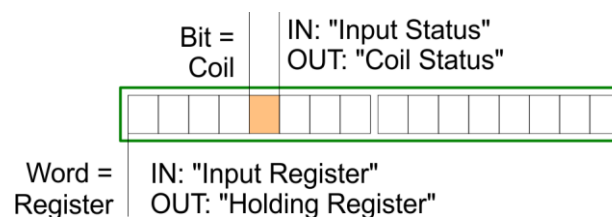
UN     M      101.3
SPB    ERR1
L      MW     102
T      MW     104
ERR1:  NOP    0
```

OB1 - Описание

1. ➤ Вызов блока обработки данных ведомого устройства Modbus RTU с корректными параметрами.
2. ➤ Если интерфейс был корректно инициализирован в OB 100, ведомое устройство готово к работе и MW102 содержит 7006h.
3. ➤ Начальный регистр Modbus для функции 05h, используемой для записи данных в образ процесса, в приведенном примере может изменяться через параметр FC5_ADDR_OUTPUT_START (слово 58 в экземплярном блоке данных).
4. ➤ Установка в ЦПУ для M101.0 значения TRUE.
⇒ Ведомое устройство Modbus теперь запущено.
5. ➤ Ведущее устройство Modbus отправляет запрос с кодом функции 03h, начальным адресом 10 и количеством 30.
⇒ Ведомое устройство в ответ передает 60 байт данных, расположенных в его памяти, начиная с адреса MB200. Значение на выходе DR устанавливается на время одного цикла ЦПУ и, соответственно, значение M101.2 становится равно "1".
6. ➤ Ведущее устройство Modbus направляет запрос с кодом функции 05h, начальным адресом 5008 и значением FF00h.
⇒ Ведомое устройство подтверждает запрос и записывает "1" на выход Q1.0. NDR устанавливается на время одного цикла ЦПУ и, следовательно, значение M101.1 становится равно "1".
7. ➤ Ведущее устройство Modbus направляет запрос с кодом функции 03h, начальным адресом 50 (не существует!) и количеством 1.
⇒ Ведомое устройство отвечает с Exception code 02h и устанавливает на выходах ERROR/STATUS соответствующие значения на время одного цикла ЦПУ. MW104 содержит значение 8383h.

4.3 Коды FKT (коды функций Modbus)**Соглашение об использовании имён**

Для Modbus существуют соглашения об именах, которые кратко перечислены ниже:



- Modbus различает доступ к битам и к словам; биты = "Coils" и слова = "Register".
- Входные биты представляются как "Состояние входа" и выходные биты - как "Состояние катушки (Coil)".
- Входные переменные с форматом слова называются "Входные регистры (Input Registers)", а выходные - "Регистры хранения (Holding Registers)".

Определение диапазона

Для адресации регистров в Modbus обычно используются диапазоны 0x, 1x, 3x и 4x. Диапазоны 0x и 1x отведены для доступа к *дискретным* (битовым) входам и выходам, а 3x и 4x - для доступа к 16-разрядным (*аналоговым*) регистрам.

Поскольку для коммуникационных процессоров VIPA нет никакой разницы между цифровыми и аналоговыми данными, применяется следующее присвоение:

0x - Дискретные выходы

Доступ с помощью функций с кодами 01h, 05h, 0Fh

1x - Дискретные входы

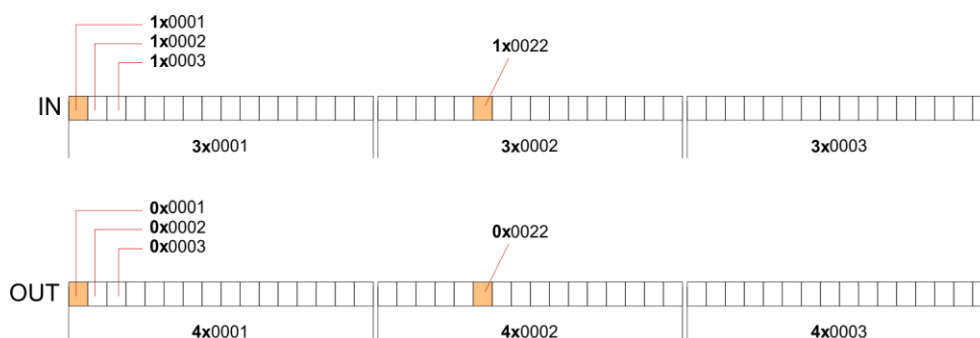
Доступ с помощью функции с кодом 02h

3x - Аналоговые входы (16-разрядные)

Доступ с помощью функции с кодом 04h

4x - Аналоговые выходы (16-разрядные)

Доступ с помощью функций с кодами 03h, 06h, 10h и 16h

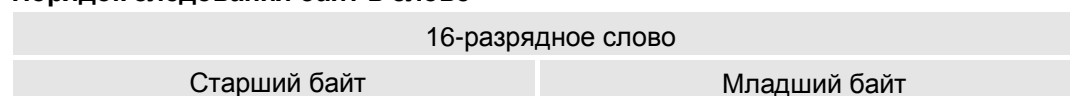


Обзор

Функции со следующими кодами позволяют ведущему устройству Modbus получить доступ к данным ведомого устройства. Описание всегда приводится относительно ведущего устройства:

Код	Команда	Описание
01h	Read n Bits	Чтение нескольких бит из области дискретных выходов 0x
02h	Read n Bits	Чтение нескольких бит из области дискретных входов 1x
03h	Read n Words	Чтение нескольких слов из области выходных регистров (регистров хранения) 4x
04h	Read n Words	Чтение нескольких слов из области входных регистров 3x
05h	Write 1 Bit	Запись одного бита в область дискретных выходов 0x
06h	Write 1 Word	Запись одного слова в область выходных регистров (регистров хранения) 4x
0Fh	Write n Bits	Запись нескольких бит в область дискретных выходов 0x
10h	Write n Words	Запись нескольких слов в область выходных регистров (регистров хранения) 4x
16h	Mask 1 Word	Запись одного слова с использованием маски в область выходных регистров (регистров хранения) 4x
17h	Write n Words and Read m Words	Запись нескольких слов (n) в область выходных регистров (регистров хранения) 4x и одновременное чтение нескольких слов (m) из области входных регистров 3x

Порядок следования байт в слове



Ответ ведомого устройства

Если ведомое устройство в ответном сообщении отправляет номер выявленной им в полученном запросе ошибки, то в качестве кода функции используется значение кода функции из запроса с установленным в нём в 1 старшим битом. Если ошибки в запросе не выявлено, в ответном сообщении код функции из запроса.

Коды FKT

Ответ ведомого	(Код функции) OR (80h)	→ Запрос с ошибкой и номер ошибки
	Код функции	→ Запрос без ошибки

Кроме того, если произошла ошибка, номер ошибки будет указан в следующем байте. Ниже приведены номера ошибок:

01h: Принятый код функции не поддерживается

02h: Адрес данных, указанный в запросе, недоступен

03h: Значение, содержащееся в поле данных запроса, является недопустимой величиной

04h: Системная шина SLIO Bus не инициализирована

07h: Ошибка общего характера

Чтение нескольких бит (функции 01h, 02h)

Функция 01h: Чтение нескольких бит из области дискретных выходов 0x.

Функция 02h: Чтение нескольких бит из области дискретных входов 1x.

Сообщение с запросом

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес первого регистра (бита данных)	Количество регистров/бит данных
x	x	0	0	0	6				
6 байт						1 байт	1 байт	1 слово	1 слово

Ответное сообщение

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Количество байт со считанными значениями	Байт 1 данных	Байт 2 данных	...
x	x	0	0	0							
6 байт						1 байт	1 байт	1 байт	1 байт	1 байт	
									252 байта (макс.)		

Чтение нескольких слов (функции 03h, 04h)

Функция 03h: Чтение нескольких слов из области выходных регистров (регистров хранения) 4x.

Функция 04h: Чтение нескольких слов из области входных регистров 3x

Сообщение с запросом

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес первого регистра	Количество регистров (слов)
x	x	0	0	0	6				
6 байт						1 байт	1 байт	1 слово	1 слово

Ответное сообщение

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Количество байт со считанными значениями	Значение слова 1	Значение слова 2	...
x	x	0	0	0							
6 байт						1 байт	1 байт	1 байт	1 слово	1 слово	
									126 слов (макс.)		

Запись значения 1 бита (функция 05h)

Функция 05h: Запись значения одного бита в область дискретных выходов 0x.

Изменение состояния бита осуществляется путём указания в поле "Состояние бита" следующих значений:

"Состояние бита" = 0000 h → Значение бита = 0

"Состояние бита" = FF00 h → Значение бита = 1

Сообщение с запросом

Заголовок Modbus/TCP							Адрес ведомого устройства	Код функции	Адрес адрес регистра (бита данных)	Состояние бита
x	x	0	0	0	6					
6 байт							1 байт	1 байт	1 слово	1 слово

Ответное сообщение

Заголовок Modbus/TCP							Адрес ведомого устройства	Код функции	Адрес адрес регистра (бита данных)	Состояние бита
x	x	0	0	0	6					
6 байт							1 байт	1 байт	1 слово	1 слово

Запись 1 слова (функция 06h)

Функция 06h: Запись одного слова в область выходных регистров (регистров хранения) 4x

Сообщение с запросом

Заголовок Modbus/TCP							Адрес ведомого устройства	Код функции	Адрес регистра	Значение регистра (слова)
x	x	0	0	0	6					
6 байт							1 байт	1 байт	1 слово	1 слово

Ответное сообщение

Заголовок Modbus/TCP							Адрес ведомого устройства	Код функции	Адрес регистра	Значение регистра (слова)
x	x	0	0	0	6					
6 байт							1 байт	1 байт	1 слово	1 слово

Коды FKT

Запись нескольких бит (команда 0Fh)

Команда 0Fh: Запись значения нескольких бит в область дискретных выходов 0x
Обратите внимание, что количество бит также дополнительно указывается в байтах, содержащих информацию для записи.

Сообщение с запросом

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес первого регистра (бита данных)	Количество регистров /бит данных	Количество байт с устанавливаемыми значениями	Байт 1 данных	Байт 2 данных	...
x	x	0	0	0									
6 байт						1 байт	1 байт	1 слово	1 слово	1 байт	1 байт	1 байт	1 байт
											248 байт (макс.)		

Ответное сообщение

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес первого регистра (бита данных)	Количество регистров/бит данных
x	x	0	0	0	6				
6 байт						1 байт	1 байт	1 слово	1 слово

Запись нескольких слов (функция 10h)

Функция 10h: Запись нескольких слов в область выходных регистров (регистров хранения) 4x.

Сообщение с запросом

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес первого регистра (слова)	Количество регистров (слов)	Количество байт с устанавливаемыми значениями	Значение слова 1	Значение слова 2	...
x	x	0	0	0									
6 байт						1 байт	1 байт	1 слово	1 слово	1 слово	1 слово	1 слово	1 слово
											124 байта (макс.)		

Ответное сообщение

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес первого регистра (слова)	Количество регистров (слов)
x	x	0	0	0	6				
6 байт						1 байт	1 байт	1 слово	1 слово

Запись 1 слова с использованием маски (функция 16h)

Функция 16h: Эта команда позволяет индивидуально сбросить или установить отдельные биты в регистре из области 4x.

Сообщение с запросом

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес регистра	Значение для наложения маски по AND (лог. И)	Значение для наложения маски по OR (лог. ИЛИ)
x	x	0	0	0	8					
6 байт						1 байт	1 байт	1 слово	1 слово	1 слово

Ответное сообщение

Заголовок Modbus/TCP						Адрес ведомого устройства	Код функции	Адрес регистра	Значение для наложения маски по AND (лог. И)	Значение для наложения маски по OR (лог. ИЛИ)
x	x	0	0	0	8					
6 байт						1 байт	1 байт	1 слово	1 слово	1 слово