SIPROTEC

Protection devices

7SJ61...7SJ64
7ST61, 7ST63
7UM61, 7UM62
7UT612, 7UT613, 7UT63
7VE6

Input/Output unit

6MD63

Communication module

Modbus
Communication profile

Revision 4.0

Edition: September 2004

C53000-L1840-C001-03

Siemens Aktiengesellschaft
C53000-L1840-C001-03

Preface

Purpose of this manual

The manual describes the functions, bus specific parameters, DIGSI parameterization and the hardware interface of the Modbus slave of the SIPROTEC devices and is divided into the following topics:

- Asynchronous communication modules → Chapter 1;
- Parameters and properties → Chapter 2;
- Data type definitions → Chapter 3;
- Modbus - Parameterization in DIGSI → Chapter 4;
- Technical data → Chapter 5.

General details about the function, operation, assembly and commissioning of the SIPROTEC devices you find in the

- SIPROTEC4 System Manual, order no. E50417–H1176–C151.

Bus mapping documentation

The following additional manuals inform you about the Modbus register map organization of the individual SIPROTEC devices:

Manual	Order number
Modbus - Bus mapping 7UM61	C53000-L1840-C005-03
Modbus - Bus mapping 7SJ61...7SJ64, 6MD63	C53000-L1840-C006-03
Modbus - Bus mapping 7UM62	C53000-L1840-C009-03
Modbus - Bus mapping 7UT612	C53000-L1840-C010-03
Modbus - Bus mapping 7UT613, 7UT63	C53000-L1840-C015-03
Modbus - Bus mapping 7ST61, 7ST63	C53000-L1840-C016-03
Modbus - Bus mapping 7VE6	C53000-L1840-C017-03

You can download these manuals

from Internet **http://www.siprotec.com**

or please contact your Siemens representative.

Modbus specification

The Modbus specification with a detailed explanation of the Modbus protocol is contained in:

- MODICON
  Modbus Protocol
  Reference Guide
  PI-MBUS-300 Rev. J
  June 1996, Modicon, Inc.

Validity

This manual is valid for the SIPROTEC devices:

- 6MD63 (firmware version 4.4 or higher),
- 7SJ61...7SJ64 (firmware version 4.4 or higher),
- 7ST61, 7ST63 (firmware version 4.0 or higher),
- 7UM61 (firmware version 4.1 or higher),
- 7UM62 (firmware version 4.0 or higher),
- 7UT612 (firmware version 4.0 or higher),
- 7UT613, 7UT63 (firmware version 4.0 or higher),
- 7VE6 (firmware version 4.0 or higher)

with Modbus communication module up to HW revision 3 and

- Modbus firmware version 02.00.05 or higher,
- Modbus firmware version 03.00.04 or higher at use of
  - the Event recorder (Sequence of Events),
- Modbus firmware version 03.01.01 or higher at use of
  - Transformer tap change commands/Transformer tap position indications (ref. to chap. 3.5 and 3.6),

with Modbus communication module from HW revision 4 and

- Modbus firmware version 04.00.04 or higher.

For device parameterization have to be used:

- DIGSI 4.21 considering the preconditions explained in chapter 4.1.3,
- DIGSI 4.3 or higher,
- DIGSI 4.4 or higher at use of
  - Transformer tap change commands/Transformer tap position indications (ref. to chap. 3.5 and 3.6),
- Modbus standard mappings 3-1 to 3-n (n = device type dependent number of standard mappings).

Additional Support

For questions regarding SIPROTEC4 devices, please contact your Siemens representative.

Training courses

Individual course offerings may be found in our Training Catalog and questions can be directed to our Training Centre. Please contact your Siemens representative.

Target audience

Protection engineers, commissioning engineers, personnel concerned with adjustment, checking and service of selective protective equipment, automatic and control facilities and personnel of electrical facilities and power plants.

Warning!

Hazardous voltages are present in this electrical equipment during operation. Non-observance of the safety rules can result in severe personal injury or property damage.

Only qualified personnel shall work on and around this equipment after becoming thoroughly familiar with all warnings and safety notices of this and the associated manuals as well as with the applicable safety regulations.

The successful and safe operation of this device is dependent on proper transport and storage, proper handling, installation, operation, and maintenance by qualified personnel under observance of all warnings and hints contained in this and the associated manuals.

In particular the general erection and safety regulations (e.g. IEC, EN, DIN, VDE, or other national and international standards) regarding the correct use of high-voltage installations must observed. Non-observance can result in death, personal injury or substantial property damage.

QUALIFIED PERSONNEL

For the purpose of this manual and product labels, a qualified person is one who is familiar with the installation, construction and operation of the equipment and the hazards involved. In addition, he has the following qualifications:

- Is trained and authorized to energize, de-energize, clear, ground and tag circuits and equipment in accordance with established safety practices.
- Is trained in the proper care and use of protective equipment in accordance with established safety practices.
- Is trained in rendering first aid.

Typographic and graphical conventions

The following text formats are used to identify concepts giving device information described by the text flow:

`Parameter names`, or identifiers for configuration or function parameters that appear in the device display or on the screen of a PC (with DIGSI) are shown in mono-script (same point size) bold text. This also applies to header bars for selection menus.

`Parameter conditions`, or possible settings of parameters that appear in the device display or on the screen of a PC (with DIGSI), are additionally shown in italic style. This also applies to selection items for selection menus.

„`Annunciations`“, or identifiers for information produced by the device or required by other devices or from the switchgear is shown in mono-script (same point size) and placed into quotation marks.

For diagrams in which the identifier type results from the representation itself, text conventions may differ from the above-mentioned.

Revision index

Listing of the changes between the editions of this manual:

Modified chapters / pages	Edition	Reasons of modification
	1.0	First edition, Doc.-No.: C53000-L1840-C001-03 Oct. 23rd, 2001
Page 4 Chap. 4.1.4 Chap. 4.4.3 Page 4-69	2.0	• Manual is also valid for 6MD63 and 7SJ61...64 devices • New chapter “Interface selection and mapping selection in DIGSI 4.3 or higher” • Description of Scaling indices 7 to 9 added • Further notes regarding allocation and processing of commands via Modbus added. Jan. 14th, 2002
Chap. 1.3	2.1	NEGATIVE_ACKNOWLEDGE = Exception code 07 March 19th, 2002
general Chap. 1.1.1, 4.1 Chap. 1.1.1 Chap. 1.1.2 Chap 1.1.3 Chap. 1.2 Chap. 1.3 Chap. 1.5 Chap. 2.6 Chap. 3.1.4 Chap. 3.3 Chap. 3.5.1	3.0	• Manual is also valid for 7UM61 devices from V4.1 • additional baud rates 38400 and 57600 with Modbus from V03.00.04 • notes to stop bits for serial communication added • new chapter: “Event recorder settings” • new chapter: “Extended bus timing settings” • max. number of registers per query: distinction between RTU and ASCII mode • Exception Code 03 if too many registers shall be read or written with one query • new chapter: “Event recorder (Sequence of Events)” • Data type “Message block for Event recorder” added • Figure 4-5 added • Figure 4-9 added • Time interval for time synchronization via Modbus has to be 1 minute Jan. 20th, 2003
general Chap. 2.5 Chap. 2.6 Chap. 1.5.1, 2.8 Chap. 3.3 Chap. 2.2	3.1	• Manual is also valid for devices 7UT613, 7UT63 from V4.0, 7ST61, 7ST63 from V4.0 and 7VE6 from V4.0 • Transformer tap change commands and transformer tap position indications are supported with Modbus firmware from V03.01.01 • new chapter: “Transformer tap change command (TC)” • new chapter: “Transformer tap position indication (TM)” • Transformer tap position indications are also available via Event recorder (Sequence of Events) • Transformer tap change commands and its corresponding transformer tap position indications have to be routed to the same Holding register • Explanation incl. examples for control of double commands using Force Multiple Coils and Force Single Coil added. Jun. 19th, 2003

Modified chapters / pages	Edition	Reasons of modification
Chap. 1 Chap. 2.1 Chap. 2.2 general	4.0	• Chap. “Asynchronous communication modules” added (new HW revision 4), *the following chapter numbers moved up* • new parity option: NONE2 • Diagnostic subfunctions 10, 12, 13, 14 are supported • Page numbering in the manual now continuous, not chapter-related any more Sept. 10th, 2004

Table of contents

Asynchronous communication modules

1

This chapter shows the hardware and software necessary for Modbus communication with SIPROTEC devices and describes the display of module-specific information at the device.

1.1 Communication module types and hardware revisions

1.1.1 Communication module types

Two communication modules are available for the connection of Modbus to the SIPROTEC devices:

RS485 bus interface

Asynchronous module with isolated RS485 interface.

This module also is called AME module (**a**synchronous communication **m**odule **e**lectrical) subsequently.

Fibre-optical bus interface

Asynchronous module with fibre-optical interface.

This module also is called AMO module (**a**synchronous communication **m**odule fibre-**o**ptical) subsequently.

Technical data

The technical data of the above-mentioned asynchronous communication modules are summarized in chap. 5.

1.1.2 Hardware revisions

There exist two different hardware revisions for asynchronous communication modules:

- up to HW revision 3:
  delivery up to the end of year 2004
- from HW revision 4:
  replacement for modules up to HW revision 3, delivery from beginning of 2005

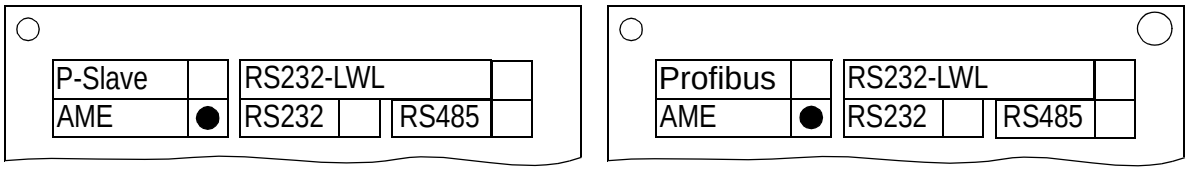
The communication modules from HW revision 4 are function compatible to the modules up to HW revision 3.

Please note the dependency of the Modbus firmware versions with the HW revisions described in chap. 1.1.3.

The hardware revision of the asynchronous communication modules is also recognizable in build-in condition at the rear of the SIPROTEC device at the labelling of the communication module mounting bracket:

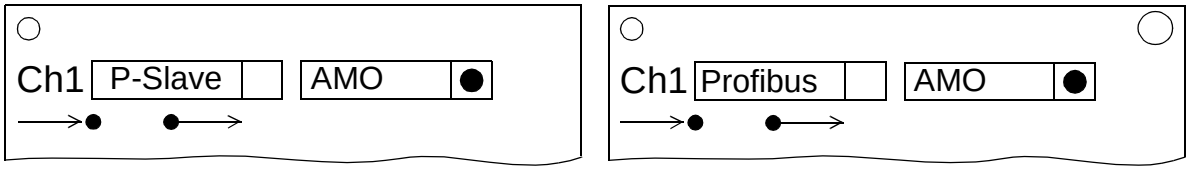
- up to HW revision 3: identification table starts with “P-Slave”
- from HW revision 4: identification table starts with “Profibus”

Asynchronous module RS485 (AME module)



up to HW revision 3 | from HW revision 4

Asynchronous module fibre-optical (AMO module)



up to HW revision 3 | from HW revision 4

Figure 1-1 HW revisions of the communication modules, labelling of the mounting brackets

General details about the assembly of communication modules as well as the setting of the terminating resistors on the AME modules you find in the SIPROTEC4 System Manual (ref. to page 3).

1.1.3 Compatibility of the communication module hardware with Modbus firmware versions and mapping files

Hardware and firmware

Please note the following listed compatibility between the hardware revisions of the communication modules and the Modbus firmware versions:

Hardware revision	Firmware version to be used
up to HW revision 3	up to Modbus firmware V03
from HW revision 4	from Modbus firmware V04

Table 1-1 Hardware revisions and firmware versions

The Modbus firmware for communication modules from HW revision 4 is:

- function compatible with firmware versions for modules up to HW revision 3 (i.e. contains all there contained functionalities),
- offers additional functionalities, e.g.:
  - display of module-specific information at the device (ref. to chap. 1.2).

Note:

If, during loading of the Modbus firmware on the communication module, a non-compatible hardware revision is recognized, then the firmware update is cancelled.

Please, in case of an abort of loading the Modbus communication firmware, check first the dependencies indicated in Table 1-1.

After attempting to load a Modbus firmware version on a non-compatible hardware revision, the SIPROTEC device remains in the loader mode (display = empty, LED 5 = ON, LED 6 flashing) and loading of a correct firmware version or an initial reset is expected.

If no firmware shall be loaded in this situation, then the device has to be switched off and (after at least 3 sec.) switched on again.
The previous firmware configuration is then used furthermore.

Hardware and mapping files

There is no compatibility reduction between Modbus mapping files of the SIPROTEC devices and the hardware revision of the communication modules, i.e.:

- the known Modbus mapping files for SIPROTEC devices, offered in DIGSI and used so far, are used for parameterization furthermore,
- existing parameterizations can be used further, even if a communication module up to HW revision 3 is replaced by a communication module from HW revision 4 (considering the firmware compatibility in Table 1-1).

1.2 Display of module-specific information at the SIPROTEC device

Note:

The following prerequisites are necessary for using the functionality “Display of module-specific information at the SIPROTEC device”:

- Asynchronous communication module from HW revision 4 with Modbus firmware from V04.00,
- SIPROTEC device firmware which supports this function, e.g.:
  - 7SJ61...7SJ64, 6MD63 device firmware from V4.50.

Changing parameters for the Modbus slave of the SIPROTEC device is exclusively possible using the DIGSI parameterization system.

If an asynchronous communication module up to HW revision 3 is used, then in a SIPROTEC device with above-mentioned firmware V4.50 the following text is displayed in case of selecting the menu item for display of module-specific information:

```
**** LIST EMPTY ****
```

If the display of module-specific information is not supported by the firmware of the used SIPROTEC device then the below-mentioned menu items are not offered for selection.

The values in the display of module-specific information are actualized every 500 ms. It is therefore possible that short-time changes of information are not displayed.

The display of module-specific information is accessible with the following menu items or buttons:

- MENU
- Test/Diagnosis → 5
- Modulinfo → 5
- Port B → 1

The maximum number of displayed information, separated in five blocks, is shown in Figure 1-2.

Depending on the selected mapping file (and the resultant functional range) block 5 is not existing.

The data in the individual information blocks are explained in the following chap. 1.2.1 to 1.2.5.

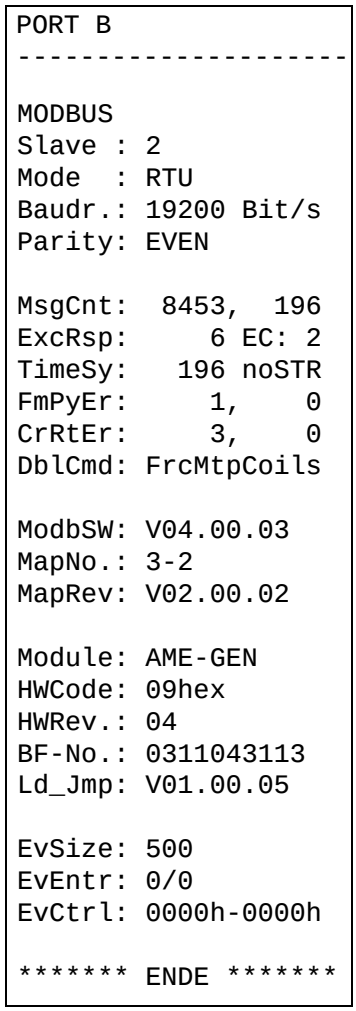


Figure 1-2 Display of module-specific information at the device

1.2.1 Block 1: Status and parameters of the Modbus slave

Block 1 of the module-specific information shows parameterization data of the Modbus slave of the SIPROTEC device (ref. to chap. 2.1.1).

Note:

If no Modbus mapping file was selected during parameterization in DIGSI then all entries of Block 1 are marked with the sign '-':

```
Slave : -
Mode  : -
...
```

Slave Display of the slave address which was entered during parameterization in DIGSI for `GlobalSection.Slave_Addr`.

Mode Display of the Modbus mode (RTU or ASCII) which was entered during parameterization in DIGSI for `GlobalSection.ModbusMode`.

Baudr. Display of the baud rate of the serial communication which was entered during parameterization in DIGSI for `GlobalSection.Baudrate`.

Parity Display of the way of parity bit evaluation which was entered during parameterization in DIGSI for `GlobalSection.Parity`.

1.2.2 Block 2: Status and diagnosis

Block 2 contains various diagnostic counters and status information.

Note:

- All diagnostic counters, which are shown followingly, are 16 bit values (data range: 0...65535) and start again with 0 after an overflow. Overflows of these counters are not signalized.
- A reset of the counters is possible via Modbus using Modbus function 08 - "Diagnostics", Subfunction 10 - "Clear Counters" (ref. to chap. 2.2).

MsgCnt Two diagnostic counters of message processing:

1. Counter 1: Quantity of messages addressed to the slave, or broadcast, that the slave has processed since last restart of the SIPROTEC device or since the last "Clear Counters" diagnostic message.
2. Counter 2: Quantity of broadcast message in the value of Counter 1.

The Counter 1 of the quantity of processed messages can also be requested via Modbus using Modbus function 08 - "Diagnostics", Subfunction 14 - "Return Slave Message Count".

Note:

If the counter of the quantity of processed messages is not incremented, then:

- the Modbus slave is not addresses by the master, because e.g.
  - the slave is disconnected from the master,
  - the master does not send messages to this slave (wrong slave address),
- the communication settings (baud rate etc.) are not consistent in the master and the slave device (the counters of framing and/or parity errors then also is incremented, see below).

If the counter of the quantity of processed messages is incremented but the Modbus master does not receive or interpret correct response messages from the slave, then:

- The Response Time-Out setting in the master device could be too low (especially for lower baud rates). In this case, master and slave send messages simultaneously on the half duplex bus (the master a new query message and the slave a response message to the previous query).
- The configured default value of the **Bus silent time RxTx in RTU mode** of the slave could be too low for the master (especially for higher baud rates, see notes in chap. 2.1.3).

ExcRsp

Diagnostic counter of the quantity of Modbus exception responses returned by the slave since last restart of the SIPROTEC device or since the last “Clear Counters” diagnostic message.

If this counter is not equal to 0 then the exception code (EC) of the last transmitted exception response is shown additionally.
The exception code display is hidden if no exception response was returned.

Supported exception codes are listed in chap. 2.3.

The example in Figure 1-2 shows, that the slave returned six exception responses since last restart of the SIPROTEC device or since the last “Clear Counters” diagnostic message, last exception response with ILLEGAL_DATA_ADDRESS (exception code 2).

The counter of the quantity of Modbus exception responses can also be requested via Modbus using Modbus function 08 - “Diagnostics”, Subfunction 13 - “Return Bus Exception Error Count”.

TimeSy

Time synchronization of the SIPROTEC device via Modbus.

This diagnostic counter shows the quantity of time synchronization messages which are passed on the SIPROTEC device since last restart of the SIPROTEC device or since the last “Clear Counters” diagnostic message.

Depending on the method of data acceptance for time synchronization (ref. to chap. 2.1.1) the time synchronization message is passed on the SIPROTEC device either by writing a value in the “Set Time and Date” register or by transferring the complete values of date and time to the corresponding Holding registers.

The selected method of data acceptance for time synchronization via Modbus is also displayed in this line of the module-specific information:

`noSTR` - "Set Time and Date" register is not used

`useSR` - "Set Time and Date" register is used

FmPyEr

Two diagnostic counters for framing and parity errors of the serial data transmission.

A **framing error** is reported by the serial communication module if a stop bit is requested after the reception of a byte frame but the level at the data line remains low. If this counter is incremented continuously, then this indicates a wrong baud rate setting.
Disconnecting and reconnecting the bus line during active communication can be the reason of single framing errors.

If the serial communication is configured with evaluation of parity bits (EVEN or ODD, ref. to chap. 2.1.1) then a **parity error** indicates a wrong value of a parity bit in the serial byte frame.
Short-time data transmission errors (e.g. because of disturbing influences) are indicated by single incrementing of this counter.
If the counter of parity errors is incremented continuously then the settings for parity bit evaluation are not consistent in the master and the slave device.

CrRtEr

Two diagnostic counters for CRC errors and response-time errors.

A **CRC error** is recognized if the result of the CRC calculation of the received Modbus message (CRC calculation in Modbus mode RTU or LRC calculation in Modbus mode ASCII) is not equal to the CRC/LRC value in the Modbus message from master. Mostly, the reason of CRC errors are data transmission errors (e.g. because of disturbing influences).

The counter of the quantity of CRC errors can also be requested via Modbus using Modbus function 08 - “Diagnostics”, Subfunction 12 - “Return Bus Communication Error Count”.

Response-time errors occur if the time for evaluating the data from the query message and for creating the response message in the slave is longer than the configured “Maximum slave response time” (ref. to chap. 2.1.1).
The value of the parameter “Maximum slave response time” should not be set less than the default setting of 10 ms.

DblCmd

For the control of double commands via Modbus there exist a parameter which determines, which Modbus function can be used (ref. to chap. 2.1).

The current setting is shown here:

- `FrcMtpCoils`
  corresponds to `GlobalSection.SingleBitDoubleCmdCtrl = 0`, i.e. only with Modbus function "Force Multiple Coils"
- `FrcSglCoil`
  corresponds to `GlobalSection.SingleBitDoubleCmdCtrl = 1`, i.e. with Modbus function "Force Single Coil" or "Force Multiple Coils" alternatively

1.2.3 Block 3: Firmware versions and mapping file

Block 3 of the module-specific information shows the Modbus firmware version as well as the number and version of the selected mapping file.

Note:

If no Modbus mapping file was selected during parameterization in DIGSI then all entries of Block 2 are marked with the text 'not loaded':

```
MapNo.: not loaded
MapRev: not loaded
```

ModbSW Modbus firmware version loaded on the communication module.

MapNo. Number of the selected standard mapping.

The mapping file determines the data size which is available via Modbus for the SIPROTEC device.
Depending on the device type, several standard mappings are offered for parameterization in DIGSI (ref. to page 3, "Bus mapping documentation" of the SIPROTEC devices).

MapRev. Version of the selected standard mapping with the number `MapNo.` (see above).

1.2.4 Block 4: Module hardware information and boot firmware version

Block 4 of the module-specific information contains hardware information for the built-in communication module.

Module Hardware type of the communication module built-in in the SIPROTEC device:

Module	Explanation	Note
PSE_GEN	PROFIBUS module RS485	not for Modbus, please replace
PSO2_GEN	PROFIBUS module fibre-optical, double loop (with two fibre-optical channels)	not for Modbus, please replace
PSO1_GEN	PROFIBUS module fibre-optical, single loop (with one fibre-optical channel)	not for Modbus, please replace
AME_GEN	Asynchronous module RS485	OK
AMO-GEN	Asynchronous module fibre-optical	OK

Table 1-2 Communication module hardware types

HWCode A hardware designation, coded on the module, in hexadecimal representation.

HWRev. Hardware revision of the communication module.

BF-No. Serial number (production number) of the communication module.

Ld_Jmp Version of the separate boot firmware part for start-up and with loader functions.

1.2.5 Block 5: Status of the event list via Modbus

Block 5 is only available if the selected mapping file supports a “Sequence of Events” recorder via Modbus (ref. to chap. 2.5) and contains information about the status of transmission of indications using this Event recorder mechanism.

EvSize Number of the parameterized (at most possible) entries in the Event recorder on the communication module (ref. to chap. 2.1.2, `GlobalSection.EvtRec_ListSize`).

EvEntr Number of entries which are stored in the Event recorder incl. the number of entries which were offered to the master with the last read message from the Holding register area of the message blocks (ref. to chap. 2.5.2, max. three entries), e.g.:

```
EvEntr: 12/3
```

Twelve entries are currently contained in the Event recorder and three entries from these were offered to the Modbus master with the last read access to the Holding register area of the message blocks. The receipt of the evaluation of these three entries in the message blocks is expected from the Modbus master.

Note:

The three areas in the Holding registers for transmission of the Message blocks count as Event recorder entries (in addition to the parameterized “Number of entries in the Event recorder”, see `EvSize`), if they contain Message blocks for reading and these were not given a receipt yet.

For that reason the first counter in the line `EvEntr` can have an at most of the value three (number of Message blocks) greater value than shown for `EvSize`.

A buffer overflow is also shown in this line for the duration of signaling of this indication to the Modbus master, e.g. (with `EvSize` equal 100):

```
EvEntr: 103/3, Ovfl.
```

If there exist entries in the event list but the number of event list entries, which are offered to the Modbus master, remains 0, then no read accesses to the Holding register area of the message blocks are received from the master or the Holding register area of the message blocks was read incompletely (ref. to chap. 2.5.2).

EvCtrl

Current contents of the handshake register “Control_I” (sent to the Modbus master, ref. to chap. 2.5.2.2) and “Control_O” (received last from the Modbus master, ref. to chap. 2.5.2.3).

The display

```
EvCtrl: 0108h-0307h
```

means e.g., that one message block with sequence number 8 is offered to the master but no receipt was received for this because the last received receipt was for three message blocks with sequence number 7.

Parameters and properties

2

This chapter describes the properties and functions of the Modbus slave and the bus specific parameters which have to be defined during parameterization of the SIPROTEC devices for Modbus communication.

2.1 Bus specific parameters

The following settings for the serial communication between the Modbus master and the Modbus slave have to be defined during parameterization of the SIPROTEC device.

Names written in `MonoScriptText` are the associated designations of the bus specific parameters in the DIGSI parameterization software (ref. to chap. 4.1).

Note:

Modbus Plus is not supported by the Modbus slave of the SIPROTEC devices.

2.1.1 Basic Modbus settings

Slave address `GlobalSection.SlaveAddress`

Permissible slave addresses are in the range between 1 and 247.

Modbus mode `GlobalSection.ModbusMode`

The Modbus slave of the SIPROTEC device supports the two serial transmission modes ASCII and RTU:

- In ASCII mode each byte in a Modbus message is sent as two ASCII characters. For error checking a Longitudinal Redundancy Check (LRC) is used.
- If the Modbus slave is configured to communicate on a Modbus network using RTU mode, each byte in a Modbus message contains two hexadecimal digits. In RTU mode a Cyclical Redundancy Check (CRC) is applied for frame checking.

Baud rate `GlobalSection.Baudrate`

The following baud rates are available:

- 300, 600, 1200, 2400, 4800, 9600 and 19200 Bit/s,

in addition with Modbus version 03.00.04 or higher:

- 38400, 57600 Bit/s.

Parity `GlobalSection.Parity`

The parity is adjustable to:

- even or odd parity bit (1 = EVEN, 2 = ODD) in ASCII mode,
- none, even or odd parity bit (0 = NONE, 1 = EVEN, 2 = ODD) in RTU mode,

additional from Modbus communication module version 04.00.04:

- no parity and two Stop bits in ASCII mode (3 = NONE2),
- no parity and two Stop bits in RTU mode (3 = NONE2).

Stop bits

Up to Modbus communication module HW revision 3:

- Always one Stop bit is used for serial communication (also if parity NONE in RTU mode is used). This setting is not changeable.

From Modbus communication module HW revision 4:

- 1 Stop bit if parity is NONE, EVEN or ODD,
- 2 Stop bits if parity is NONE2.

Maximum slave response time

`GlobalSection.MaxMSecSlaveResponseTime`

The maximum response time determines the time interval within which the Modbus slave may respond to enquiries from the master.

This value is indicated into milliseconds and must be coordinated with the time-out of the Modbus master.

The following formula is valid:

$$T_{bus} < (T_{max} + T_{bus}) < T_{master}$$

T_{bus} - Transmission time of the slave response on the bus line,
T_{max} - Maximum slave response time (default value = 10 ms),
T_{master} - Time-out of the Modbus master.

The value of the parameter “Maximum slave response time” should not be set less than the default setting of 10 ms.

Processing of broadcast messages

`GlobalSection.CoilBroadcastMsg`
`GlobalSection.HoldingBroadcastMsg`

If one of the Modbus messages "Force Single Coil", "Preset Single Register", "Force Multiple Coils" or "Preset Multiple Regs" (ref. to chap. 2.2) is transmitted from the Modbus master to the Modbus slaves using slave address 0 then all Modbus slaves recognize this message as a broadcast message and process it.

For every Modbus slave of a SIPROTEC device it can be decided whether broadcast messages are accepted for Coil Status registers and/or Holding registers.

Per default these options are enabled and all broadcast messages are processed (`CoilBroadcastMsg` = 1, `HoldingBroadcastMsg` = 1).

Method of data acceptance for time synchronization

`GlobalSection.UseSetTimeAndDateReg`

Two methods of data acceptance for time synchronization of the SIPROTEC device via Modbus are possible:

1. acceptance immediately after writing of date and time using Modbus function “Preset Multiple Regs” as a single (in general broadcast) message to the defined positions in the Holding registers or
2. the values of date and time which are transferred with a single or with separate messages to the SIPROTEC device are accepted in the moment, a value of $FFFF_{hex}$ is written in the “Set Time and Date” register.

Option 1 is activated per default (`UseSetTimeAndDateReg` = 0).

Ref. to chap. 4.5 for further information about parameter settings for time synchronization via Modbus.

Control of double commands

`GlobalSection.SingleBitDoubleCmdCtrl`

For control of double commands (ref. to chap. 3.2) can be used:

1. only Modbus function “Force Multiple Coils” (both Coil Status registers of the double command have to be written at once) or
2. Modbus functions “Force Single Coil” or “Force Multiple Coils” alternatively (the Coil Status registers can be written as single registers for switching ON and OFF the double command).

Option 1 is activated per default (`SingleBitDoubleCmdCtrl` = 0).

Signalling of “Data invalid”

`GlobalSection.ExceptionMsgAtInvalidData`

In situations when a large amount of data have to be processed with high priority (e.g. because of multiple protection pickups and TRIPs) it can happen, that the information which are offered via Modbus are invalid (not updated) for a short period of time.

This condition is recognized in the SIPROTEC device and is signaled to the Modbus master, optional with one of the following two methods:

1. A bit in the Diagnostic register (ref. to Modbus function “Diagnostics”, subfunction 2) or in the corresponding Holding register 129 (ref. to Modbus function “Read Holding Registers”) indicates “Data invalid” with the value set to 1.
   The position of this bit is described in the bus mapping documentations of the individual SIPROTEC device types (ref. to page 3).
2. A query for reading Coil Status registers, Input Status registers or Holding registers is responded with Modbus exception 06 (SLAVE_DEVICE_BUSY) in case of “Data invalid”.
   Additional the “Data invalid” situation can be recognized by reading the Diagnostic register using Modbus function “Diagnostics”, subfunction 2, and evaluation of the associated “Data invalid” bit.
   Reading of measured values (Input registers) and writing of values (commands via Coil Status registers) is possible furthermore.

Option 1 is activated per default (`ExceptionMsgAtInvalidData` = 0).

2.1.2 Event recorder settings

The following settings are only offered if a mapping file is selected which supports an Event recorder via Modbus for the used SIPROTEC device.

Please ref. to chap. 2.5 for further information regarding the properties and reading the Event recorder.

Number of entries in the Event recorder

`GlobalSection.EvtRec_ListSize`

This parameter determines the number of Event recorder entries on the communication module (range: 10 to 1000).

If the Event recorder already contains the set number of entries (as e.g. no retrieval by the Modbus master was carried out) when a new entry arises, then the “First in - First out” principle applies, i.e. the oldest entry is replaced by the new entry and an overflow flag which is transmitted to the Modbus master the next time entries are retrieved is set (ref. to chap. 2.5.2.2).

The default setting value is 500.

Record start-up General scan

`GlobalSection.EvtRec_StartupGS`

At device start-up a General scan (update of the bus data replica) is carried out between the SIPROTEC device and the communication module.

With this parameter it can be decided if these start-up values of the objects, which are parameterized for recording, are entered in the Event recorder.
Each of this entries then will be indicated in the Event recorder with an additional General scan bit (ref. to chap. 3.8).

This option is disabled by default (`EvtRec_StartupGS` = 0).

Transmit message causes

`GlobalSection.EvtRec_MessageCauses`

Selection, if message causes (especially for switching device operations: CO+/CO-, FB+/FB- etc., ref. to chap. 3.8) are recorded for every event.

Because of some message causes not result in an value change, fewer events are recorded if disabled but the Modbus master has to be able to handle message causes if enabled.

This option is disabled by default (`EvtRec_MessageCauses` = 0).

2.1.3 Extended bus timing settings

The following bus timing settings are used to adapt to special communication environments and are only available with mapping files which require Modbus version 03.00.04 or higher.

The default values of these bus timings correspond to the Modbus specification (ref. to page 3) and they don't need to be changed in general.

It can be necessary however, that e.g. if RS232/RS485 converters with RTS controlled data direction switching from the Modbus master are used, a specified minimum data direction switchover time is to satisfied and this may not be possible with higher baud rates and the bus timing default values.

Bus silent time RxTx in RTU mode

`GlobalSection.RtuSilentTime`

Minimum time interval (in bit times) for RTU mode between reception (Rx) of a query message from Modbus master until sending (Tx) the response message.

The default value is 39 bit times which corresponds to 3.5 character times (11 bits per character in RTU mode).

The minimum value is 22 bit times.
If less than 22 bit times are parameterized, the value is set to 22 bit times internally.

Bus silent time RxTx in ASCII mode

`GlobalSection.AsciiSilentTimeRxTx`

Minimum time interval (in 100 µs) for ASCII mode between reception (Rx) of a query message from Modbus master until sending (Tx) the response message.

The default value is 0, i.e. immediate answer from the Modbus slave.

Note:

The bus timings have to meet further requirements:

- RTU mode
  - Following the last transmitted character of the response message from the Modbus slave of the SIPROTEC device, the Modbus master has to ensure a bus silent time interval of at least 2 character times (i.e. 22 bit times).
  - After sending a broadcast message, the bus silent time interval has to be at least 2 character times.
    Because no responses are sent by the Modbus slaves for broadcast messages and the Modbus slaves still need a certain time for processing the query from the master, the time interval up to sending the next message has to be extended (in addition to the 2 character times) by approx. 1 ms up to max. the value of “Maximum slave response time” (ref. to page 25).
  - All bytes of the Modbus messages must be transmitted as a continuous stream. The max. silent interval between two bytes of a message is 2 character times. If the silent interval is greater than 2 character times, the message is assumed as completed and will be evaluated by the Modbus slave.
- ASCII Modus
  - The max. silent interval between two bytes of a message is 1 second.

2.2 Supported Modbus functions

Function code	Function name	Description	Broadcast[1] supported?
1	Read Coil Status (0X references)	Reading one or several Coil Status registers of the Modbus slave. A maximum of 1970 registers in RTU mode or 960 registers in ASCII mode can be read with one message. The Coil Status registers reflect the ON/OFF status of discrete outputs of the SIPROTEC device.	no
2	Read Input Status (1X references)	Reading one or several Input Status registers of the Modbus slave. A maximum of 1970 registers in RTU mode or 960 registers in ASCII mode can be read with one message. The Input Status registers reflect the ON/OFF status of discrete inputs and the status of the protection function of the SIPROTEC device.	no
3	Read Holding Registers (4X references)	Reading one or several Holding registers of the Modbus slave. A maximum of 125 registers in RTU mode or 60 registers in ASCII mode can be read with one message. The Holding registers contain device status information, mean values of measured values, metered measurands and others.	no
4	Read Input Registers (3X references)	Reading one or several Input registers of the Modbus slave. A maximum of 125 registers in RTU mode or 60 registers in ASCII mode can be read with one message. The Input registers contain recorded measured values.	no
5	Force Single Coil (0X references)	Writing (force to ON or OFF) one Coil Status register (and binary output of the SIPROTEC device assigned with that). Use function code 15 to force multiple Coil Status registers.	yes
6	Preset Single Register (4X references)	Function presets a value into a single Holding register. Use function code 16 to preset multiple Holding registers.	yes
7	Read Exception Status	This function responses the value of the eight Coil Status registers 257..264 as Exception Status to the Modbus master.	no
8	Diagnostics	This function provides diagnostic values to the Modbus master. • Subfunction 0: The data passed in the data field of the query message to the slave are returned (looped-back) in the response. • Subfunction 2: The contents of the Diagnostic register is returned in the response to the master. For this, the contents of the Holding register 129 is used. • Subfunction 10 deletes all diagnostic counters; the Diagnostic register is not deleted.[2] • Subfunctions 12, 13, 14: diagnostic counters (ref. to chap. 1.2.2)[2]	no
15	Force Multiple Coils (0X references)	Writing (force to ON or OFF) one or several Coil Status registers (and binary outputs of the SIPROTEC device assigned with these). A maximum of 1970 registers in RTU mode or 960 registers in ASCII mode can be written with one message.	yes
16	Preset Multiple Regs (4X references)	Function presets one or several Holding registers. A maximum of 125 registers in RTU mode or 60 registers in ASCII mode can be written with one message.	yes

Table 2-1 Supported Modbus functions

1 Broadcast messages from Modbus master to the Modbus slaves using slave address 0 in the Modbus message (ref. to paragraph "Processing of broadcast messages" in chap. 2.1).

2 Available from Modbus firmware version 04.00.04.

2.3 Exception responses of the Modbus slave

If the Modbus slave receives a query from the Modbus master which cannot be processed (e.g. a request to read a non-existent register), then the slave answers with an exception response message.
The following exception codes are transferred in a exception response message to the Modbus master by the Modbus slave of the SIPROTEC device:

Exception code 01 ILLEGAL_FUNCTION

The function code used in the query by the Modbus master is not supported by the Modbus slave of the SIPROTEC device (ref. to chap. 2.2 for supported functions).

Exception code 02 ILLEGAL_DATA_ADDRESS

The Modbus master addresses a register in the query for which:

- no mapping entry exist (i.e. a non-existent register),
- a single access is not allowed because the addressed register is part of a complex bus object which uses more than one register and can be read only completely.

Exception code is used furthermore, if:

- a write access to the “Set Time and Date” register for time synchronization is detected but the acceptance of time and date is configured without “Set Time and Date” register (ref. to chap. 4.5) or
- the acceptance of date and time for time synchronization is configured without “Set Time and Date” register but the date and time structure in the Holding registers is not written completely at ones (ref. to chap. 4.5).

Exception code 03 ILLEGAL_DATA_VALUE

- The Modbus master tried to write to a register for which only read access is permitted.
- Writing of an invalid value to the “Set Time and Date” register (ref. to chap. 4.5).
- The Modbus master wants to read or write more registers with one query than allowed (ref. to chap. 2.2).
- Receipt / write access to the register “SOE_Control” with an invalid sequence number (ref. to chap. 2.5.2.3).

Exception code 06 SLAVE_DEVICE_BUSY

- The Modbus slave has no valid mapping data or the Modbus registers still have not been initialized and enabled by the SIPROTEC device (after initial start or restart of the device).
- Signalling of “Data invalid” can be done by responding a query by a Modbus exception 06 (ref. to chap. 2.1, Signalling of “Data invalid”).

Exception code 07 NEGATIVE_ACKNOWLEDGE

If in a Diagnostic query (Modbus function code 8) a not supported subfunction is requested, then this is rejected with NEGATIVE_ACKNOWLEDGE (ref. to chap 2.2 for a list of supported subfunctions).

2.4 Annunciations to the Modbus master

Note:

When analyzing the annunciations of the SIPROTEC device in the Modbus master, it should be noted that due to the cycle period of the Modbus system (period between two following queries of the same data of the Modbus slave) temporary changes of an annunciation's value (ON and OFF within one cycle) may eventually not be recognized.

This applies in the first place for protection annunciations.

Protection pickup

Protection annunciations which indicate the status **"Protection pickup"** are active only for the period of time of the protection pickup.

Protection TRIP

The parameter **Minimum Duration of TRIP Command** (parameter address = 210) allows setting of the minimum duration of the TRIP command.

This time setting applies to all protection functions which may cause a TRIP signal. After a protection TRIP, the corresponding protection annunciations transmit the value ON for the programmed minimum time duration.

Note:

With the Event recorder option, additionally to the cyclic query of the present values of the bus objects, every change of an annunciation can be recorded in the Event recorder on the communication module and transmitted to the Modbus master for further evaluation, but this needs special support by the Modbus master parameterization/software (ref. to chap. 2.5).

2.5 Event recorder (Sequence of Events)

Attention!

1. For using the Event recorder the following is required:
   - Modbus communication module firmware from V03.00.04,
   - a Modbus mapping file that supports the Event recorder option.
2. Please check in the Modbus bus mapping documentation (ref. to page 3) of the SIPROTEC device that you are applying whether this device provides an Event recorder.
3. To retrieve and evaluate the annunciations from the Event recorder, the applied Modbus master must be programmed/set in accordance with the described handshake mechanism (ref. to chap. 2.5.4) and data type definition (ref. to chap. 3.8).
   The required program sequences must be generated by the user.
4. The Event recorder via Modbus is not identical to the operational event log in the SIPROTEC device, but is separately maintained on the communication module.

No annunciations are routed per default to the Event recorder in the mapping files for the SIPROTEC devices.
After selecting a mapping file, every annunciation which should be transmitted with the Sequence of Events mechanism has to be added to the Event recorder separately using the DIGSI parameterization software (ref. to chap. 4.3, Figure 4-9).

The annunciation “Data invalid” (ref. to chap. 2.1.1, Signalling of “Data invalid”) represents a particularity for the Event recorder.
“Data invalid” is generated on the communication module and has no equivalent to an object in the SIPROTEC device.

It is treated as follows:

- If the bit in the Diagnostic register and not a response with Modbus exception 06 is used for signalling the “Data invalid” situation (default option, ref. to chap. 2.1.1), then this annunciation is always routed to the Event recorder per default.
- “Data invalid” is not enclosed in General scan cycles (neither at start-up General scan).
  Use Modbus function “Diagnostics”, subfunction 2, to read the current value of this information, if necessary.

2.5.1 Properties of the Event recorder

The Modbus communication module has an Event recorder with entries of the type “Message block” (ref. to chap. 3.8).

The size of the event buffer is set with the parameter **`Number of entries in the Event recorder`** (ref. to chap. 2.1.2).

Characteristic

- The Event recorder is a ring buffer.
- In the case of buffer overflow, the oldest entry is replaced by the current entry. The Modbus master is informed about a buffer overflow with a bit in the handshake register “SOE_Control” (ref. to chap. 2.5.2.2).
- Events of the type single-point indication and double-point indication as well as transformer tap position indications, statistic values and fault currents are accepted in the Event recorder.
- Statistic values and fault currents are transferred via the Event recorder as signed 16-bit values. The value 8000_{hex} is read in case of value-overflow.
- Cyclical information (measured values, metered measurands) are not entered in the Event recorder.
- Only those annunciations of the SIPROTEC device **that are routed in DIGSI to Modbus register positions and additionally to the Event recorder** are entered in the event buffer if they change (ref. to chap. 4.3, Figure 4-9).
- Changes of the value of an object are stored with a time stamp and an identification of this object in the Event recorder (ref. to chap. 3.8).
- Additionally, message causes (CO+/CO-, FB+/FB- or similar) can be recorded.
- Entries in the Event recorder are not made for status changes (blocked, test mode etc.) if no value change is combined with that.
- Following initial or reboot there are (depending on parameterization) two possibilities (ref. to chap. 2.1.2):
  - the Event recorder is empty,
  - start-up values of the objects, which are parameterized for recording, are entered in the Event recorder, marked with a General scan bit.
- In case of failure of the communication, the event buffer is not erased. Entries are still saved and can be read by the Modbus master once the communication link is re-established (buffer overflow is indicated if necessary).
- The entries of the Event recorder (Message blocks) can be read by the Modbus master via Holding registers and must be acknowledged (ref. to chap. 2.5.4).
- After reading and acknowledging by the Modbus master, the transferred Event recorder entries are deleted from the event buffer.

2.5.2 Structure of the Holding registers for the Event recorder

This chapter describes the Holding registers for reading and acknowledging the Event recorder entries.

The specific start of this range in the Holding registers for the particular SIPROTEC device should be checked in the corresponding bus mapping documentation (ref. to page 3).

The range in the Holding registers for reading and acknowledging the Event recorder entries consist of:

- one register “No. of Event recorder entries” (read-only access),
- one handshake register “SOE_Control” (read/write access),
- three entries of the data type “Message block” (24 Holding registers in summary) for reading the Event recorder entries (read-only access).

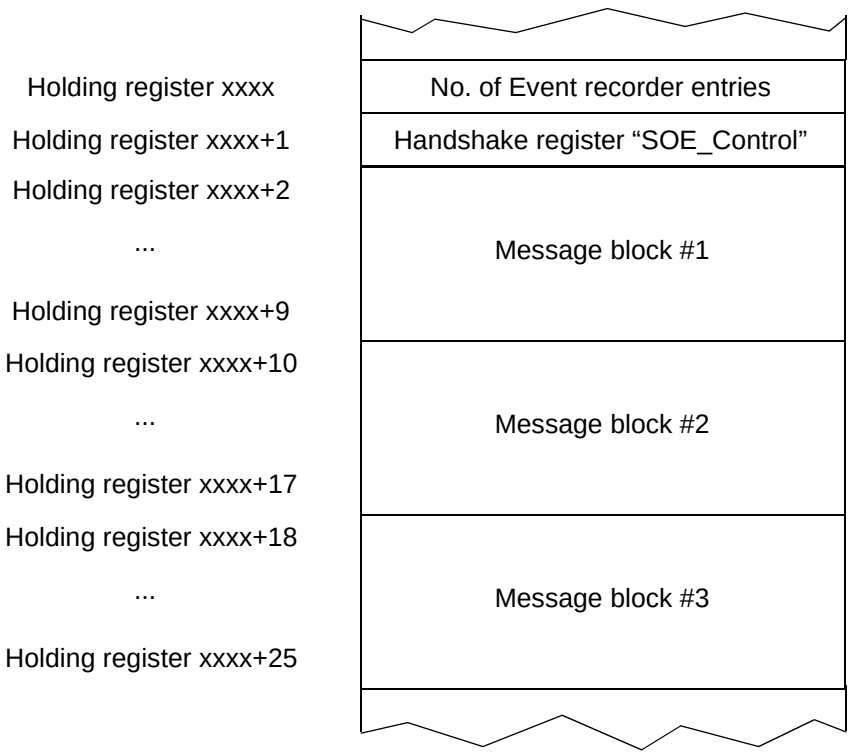


Figure 2-1 Event recorder: Structure of the associated Holding registers’ range

Note:

Only the following access to the Event recorders range in the Holding registers (ref. to Figure 2-1) is permitted:

- “SOE_Control” (read/write),
- “SOE_Control” and Message block #1 (read-only),
- “SOE_Control”, Message block #1 and Message block #2 (read-only),
- “SOE_Control”, Message block #1, Message block #2 and Message block #3 (read-only).

Additionally, the register “No. of Event recorder entries” can be read.

All other accesses to this range in the Holding registers or reading a Message block structure incompletely are responded with Modbus exception code 02 (ILLEGAL_DATA_ADDRESS).

2.5.2.1 Register “No. of Event recorder entries”

This register contains the number of Event recorder entries in the event buffer and the Holding registers which still were not read.

Range of values

- Minimum value:
  0 (no entry at present in the Event recorder)
- Maximum value:
  “Number of entries in the Event recorder” according to parameterization (ref. to chap. 2.1.2) plus three

The three areas in the Holding registers for transmission of the Message blocks count as Event recorder entries (in addition to the parameterized “Number of entries in the Event recorder”), if they contain Message blocks for reading and these were not given a receipt yet.

2.5.2.2 Handshake register “SOE_Control” (read access)

The individual bits of the handshake register “SOE_Control” have for read access the meaning indicated in Figure 2-2.

Handshake mechanism are referred to in chapter 2.5.4.

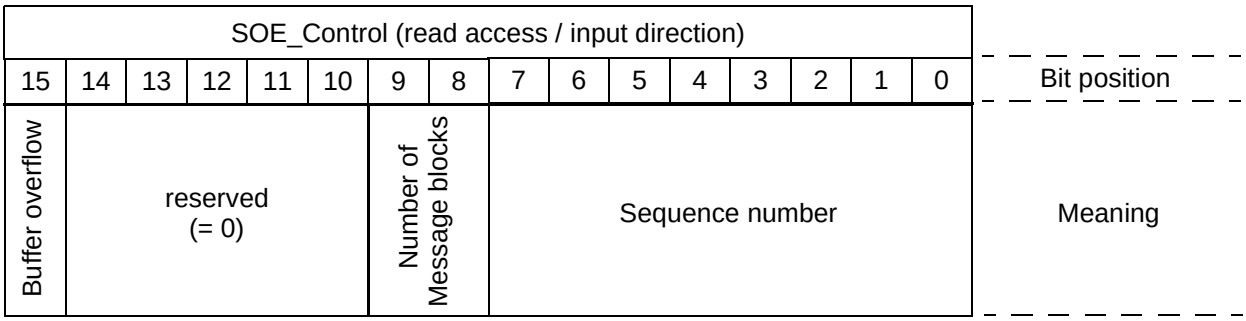


Figure 2-2 Event recorder: Handshake byte “SOE_Control” (read access)

Sequence number (Bits 0 to 7)

- With each transmission of Message blocks (entries from the Event recorder, max. three entries per query), the Sequence number is increased sequentially by 1, starting with 1 (00000001_{bin}) up to 255 (11111111_{bin}) and thereafter starting with 1 again.
- The Sequence number only has the (initial) value 0 if, following the initial or reboot of the SIPROTEC device, no entries are available in the Event recorder yet.
  For transmission of the first entry the Sequence number is incremented by 1 and then never assumes the value 0 again (unless another initial or reboot takes place).
- If no annunciation changes are to transmit during a number of bus cycles (master queries) then the Sequence number remains unchanged in its value during this time.
  The Modbus master then reads repeatedly the same last transmitted Sequence number until new entries in the Event recorder effect changes in the Message blocks, thereby incrementing the Sequence number.

At the same time, the value of the Sequence number provides the acknowledgement that the Modbus master has evaluated the read entries by writing-back the value of the Sequence number to the “SOE_Control” handshake register.

Number of Message blocks (Bits 8 and 9)

There are three Message blocks for transmission of Event recorder entries available.

The “Number of Message blocks” indicates, how many of them contain valid entries.

If none or less than three events/entries are to transmit, then the “Register type” in the unused Message blocks has the value $FF_{hex} = 255_{dec}$ (ref. to chap. 3.8).

reserved (Bits 10 to 14)

The bits of the "SOE_Control" register in input direction indicated as "reserved" are not used at present.
At these positions the value = 0 is transmitted

Buffer overflow (Bit 15)

A set bit indicates an overflow of the event buffer.

The “Buffer overflow” bit is reset following acknowledgement of the current transmitted Message blocks if no new buffer overflow occurred in the meantime.

2.5.2.3 Handshake register “SOE_Control” (write access)

The individual bits of the handshake register “SOE_Control” have for write access the meaning indicated in Figure 2-3.

Handshake mechanism are referred to in chapter 2.5.4.

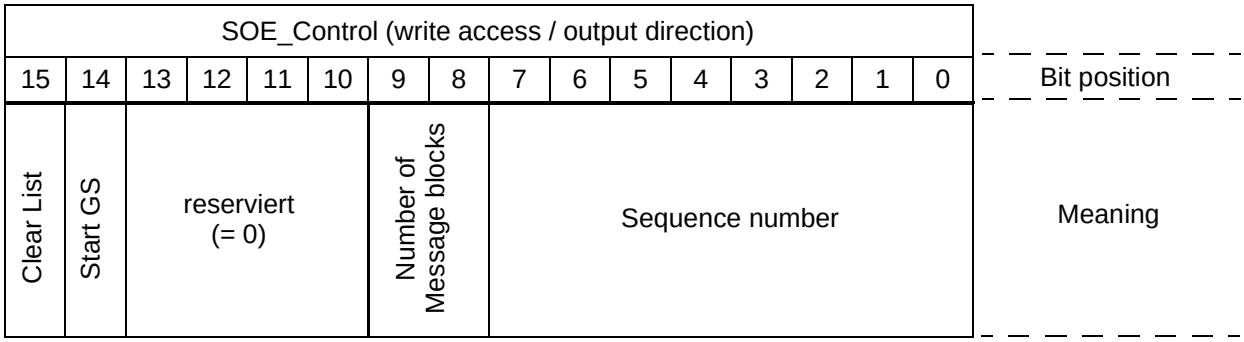


Figure 2-3 Event recorder: Handshake byte “SOE_Control” (write access)

Sequence number (Bits 0 to 7)

Acknowledgement of reception an evaluation of the read Message blocks.

The Modbus master copies the Sequence number which was read together with the Message blocks from the handshake register “SOE_Control” (input direction) as a receipt with a write access back into the “SOE_Control” register.

As long as the Sequence number was not mirrored back correctly by the Modbus master to the “SOE_Control” register, the contents (Message blocks) read last are given back at the following read accesses furthermore.

If a Sequence number was already given for a receipt, then further write accesses (without a renewed read access) are only possible with the Sequence number read last or with Sequence number = 0.
The “Number of Message blocks” is not evaluated at these write accesses.

An acknowledgment/write access with a faulty Sequence number (i.e. unequal with the one read) is answered with Modbus exception 03 (ILLEGAL_DATA_VALUE).

Write accesses with Sequence number = 0 are always possible.
Then only the command bits "Clear List" and "Start GS" are evaluated and the commands are executed, if the associated bits are set.

Number of Message blocks (Bits 8 and 9)

Reception of the number of the read Message blocks.

Together with the valid Sequence number the Modbus master has to tell the Modbus slave how many of the Event recorder entries, provided in the three Message blocks of the Holding registers, were read and evaluated.

The following reactions are defined as receipt:

Receipt value in “No. of Message blocks”	Reaction
= 0	The present Message blocks are offered for reading furthermore, the Sequence number is incremented. If less than three Message blocks were offered when the Sequence number was read last and in the meantime further entries are entered in the Event recorder, then it is filled up on the max. three possible Message blocks.
less than the value “No. of Message blocks” read from “SOE_Control” register	The number of a receipt given Message blocks are deleted from the event buffer. Not readed Message blocks are moved up in the Holding registers’ range for Message blocks and the remaining of these Holding registers are filled up with new entries from the Event list for reading, if necessary. The Sequence number is incremented.
equal to the value “No. of Message blocks” read from “SOE_Control” register	All read Message blocks are deleted from the event buffer. If there are even further entries in the Event recorder then they are now offered for reading and the Sequence number is incremented.
greater than the value “No. of Message blocks” read from “SOE_Control” register	is treated as: equal to the value “No. of Message blocks” (see above)

reserved (Bits 10 to 13)

The bits of the "SOE_Control" register in output direction indicated as "reserved" are not used at present and are not been evaluated by the Modbus slave.
At these positions the value = 0 should be transferred.

Start GS (Bit 14)

Setting the bit “Start GS” triggers one General scan cycle.

During a General scan cycle the values of the annunciations which are parameterized for recording in the Event recorder are explicitly (sequentially) requested from the SIPROTEC device with their present values and entered in the Event recorder with an General scan bit (ref. to chap. 3.8).

The “Start GS” bit can be transferred/set by the Modbus master with:

- the next receipt of read Message blocks to the Modbus slave or
- writing “SOE_Control” and Sequence number = 0 (if no receipt is to give).

If “Start GS” is set before the end of a just running Generals scan cycle then the command is ignored (the running General scan cycle will be finished, no re-triggering).

Therefore, between two General scan requests must be a corresponding time distance (or requests in fault situations only).

A set bit “GS End” in the byte “Indication type” of the last annunciation belonging to the General scan cycle indicates the end of the General scan cycle (ref. to Figure 3-9).

Clear List (Bit 15)

Setting the bit “Clear List” deletes the Event recorder.

All entries which are contained in the Event recorder at this time are lost.
The register “No. of Event recorder entries” is set to 0.

The bit can be transferred/set by the Modbus master with:

- the next receipt of read Message blocks to the Modbus slave or
- writing “SOE_Control” and Sequence number = 0 (if no receipt is to give).

Deleting the Event recorder does not cause a reset of the Sequence number.
The next (incremented) Sequence number is used when the next Message blocks are to transmit.

If at reception of "Clear List" the Message blocks read last still are not given a receipt, then this does not need to be carried out any more after deleting the Event recorder.

Start GS and Clear List simultaneous

If the command bits "Start GS" and "Clear List" are set simultaneously in the “SOE_Control” write message then first “Clear List” is executed and after that “Start GS”.

2.5.3 Message blocks

Via these Holding registers the entries of the Event recorder are transferred (max. three Message blocks with one Modbus message).

The information in the handshake register “SOE_Control” show, how many Message blocks are offered for reading in the Holding registers (ref. to chap. 2.5.2.2, “Handshake register “SOE_Control” (read access)”).

The data type “Message block” is described in chap. 3.8.

2.5.4 Handshake mechanism

The Modbus master polls (reads cyclically) the register “SOE_Control” to determine, if the Event recorder contains entries.

An incremented “Sequence number” in the “SOE_Control” register indicates new entries which are ready for reading (ref. to chap 2.5.2.2).

If new entries in the Event recorder are recognized, the Modbus master reads the “SOE_Control” register again and simultaneously max. three Message blocks.

The max. number of Message blocks in the Holding registers which contain valid data is indicated in “Number of Message blocks” of the previous read “SOE_Control” register (ref. to Figure 2-2).

If Message blocks were already read during polling of “SOE_Control” then these can be evaluated first.

After evaluation of the Message blocks, the Modbus master gives a receipt to the Modbus slave by writing the read Sequence number and the number of evaluated Event recorder entries back to the “SOE_Control” register.

The Modbus slave now deletes the read and evaluated entries from the Event recorder and increments the Sequence number, if further entries have to be read.

If a communication failure appears (CRC error or the like) then the master has to repeat the read access.
As long as no receipt is given with a write access to the “SOE_Control” register, neither Event list data are changed in the Holding registers nor deleted.

The above described mechanism of polling, reading, evaluating and giving a receipt now continues.

If the register “No. of Event list entries” is read with the register “SOE_Control” simultaneously, then it can be estimated, if and how many Message blocks should be read with the next query to optimize read accesses.
This results from the difference of “Number of Event recorder entries” and the number of valid Message blocks according the “SOE_Control” register.

Data type definitions

3

This chapter describes the data types which are used for variables in the Modbus registers.

Note:

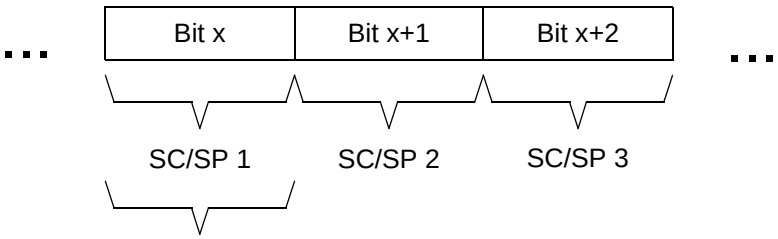
Depending on the SIPROTEC device not all of the indicated data types are used.

Please refer to the related bus mapping documentation (see page 3) for details about the Modbus register map organization of the respective SIPROTEC device.

3.1 Single command (SC) / Single-point indication (SP)

Range of values 0 - OFF

1 - ON



Coil / Input Status register or one bit of a Holding register

Bild 3-1 Data type: Single command / Single-point indication

3.2 Double command (DC) / Double-point indication (DP)

Range of values

0 (bit 1 = 0 and bit 0 = 0) - “Not applicable" for DP, not permissible for DC

1 (bit 1 = 0 and bit 0 = 1) - OFF

2 (bit 1 = 1 and bit 0 = 0) - ON

3 (bit 1 = 1 and bit 0 = 1) - Intermediate position for DP, not permissible for DC

Note:

- Depending on the type of the double-point indication, which was selected in DIGSI, it is transmitted:
  - Type DM: 0 = “Not applicable”, 3 = Intermediate position “00” or “11”,
  - Type DM_S: 0 = “Not applicable” or Intermediate position “00”
    3 = Intermediate position “11”
- "Not applicable":
  Double-point indication is not configured (not assigned to a binary input).

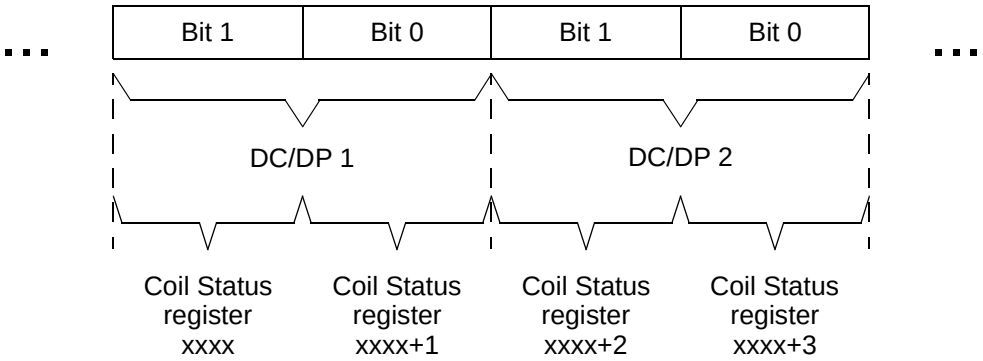


Bild 3-2 Data type: Double command / Double-point indication

Attention!

The data type double command in this chapter presupposes that the associated checkback indication is parameterized as a double-point indication.

A double command with a single-point indication as checkback indication or without checkback indication is controlled via Modbus like a single command (ref. to chap. 3.1), because the treatment of a double command via Modbus depends on the type of the associated checkback indication.

The parameter **`Control of double commands`** (ref. to chap. 2.1) determines the Modbus functions which can be used for transmission of double commands in the output messages.

Force Multiple Coils

Both Coil Status registers of the double command (ref. to Figure 3-2) must be contained in the “Force Multiple Coils” message from the Modbus master to the Modbus slave.

Example:

The Circuit breaker is routed to Coil Status registers 1 and 2.

For switching the Circuit breaker ON, the Coil Status register 1 is set to 1 and the Coil Status register 2 is reset to 0, both simultaneously using a “Force Multiple Coil” message:

Function code	0F	(= 15_{dec})
Starting Address (2 Bytes)	00 00	(= register number -1)
Quantity of Registers (2 Bytes)	00 02	
Byte count	01	
Registers Value	01	(= 000000**01**$_{bin}$)

The relation between register numbers and the register addresses in the Modbus message is explained in chap. 4.2.

Force Single Coil

OFF command:
Set the value 1 in the Coil Status register which contains the Bit 0 of the double command (ref. to Figure 3-2).

ON command:
Set the value 1 in the Coil Status register which contains the Bit 1 of the double command (ref. to Figure 3-2).

Only “Force Single Coil” messages with the value for setting the Coil Status registers to 1 (output value in the Modbus message = FF 00_{hex}) are to transfer.

Example:

The Disconnect switch is routed to Coil Status registers 3 and 4.

For switching the Disconnect switch OFF, the Coil Status register 4 is set to 1 using a “Force Single Coil” message:

Function code	05	
Output Address (2 Bytes)	00 03	(= register number -1)
Output Value	FF 00	

Switching the Disconnect switch OFF:

Function code	05	
Output Address (2 Bytes)	00 02	(= register number -1)
Output Value	FF 00	

3.3 Measured value (signed integer)

Range of values -32768 to +32767

(-32768 = “Overflow” or “Invalid”)

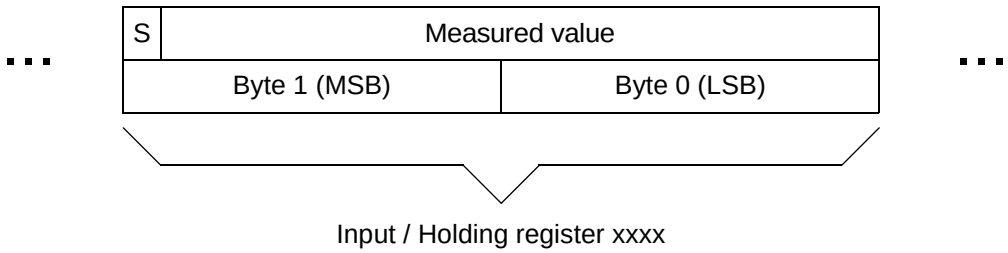


Bild 3-3 Data type: Measured value (signed integer)

Status bit (S)

- Status bit corresponds to the sign bit, active if negative measured value.

  Negative measured values are transmitted in the two's complement, i.e.:
  -1 = $FFFF_{hex}$, -2 = $FFFE_{hex}$, ..., -32767 = 8001_{hex}
- Status bit = 1 and measured value = 0
  (i.e. transmission of the value 8000_{hex} = -32768):
  Measured value overflow or invalid measured value.

Note:

The value -32768 for signalling of "Overflow" or "Invalid" is only used for measured values in input direction.

If an evaluation of the status of a measured value in output direction[1] is required in the SIPROTEC device then a separate information has to be used for this.

1 Measured values in output direction (Modbus master writes to a measured value in the SIPROTEC device) are available for special features in some device types only (e.g. 7UM62).

3.4 Metered measurand (unsigned long)

Range of values 0 to +4294967295

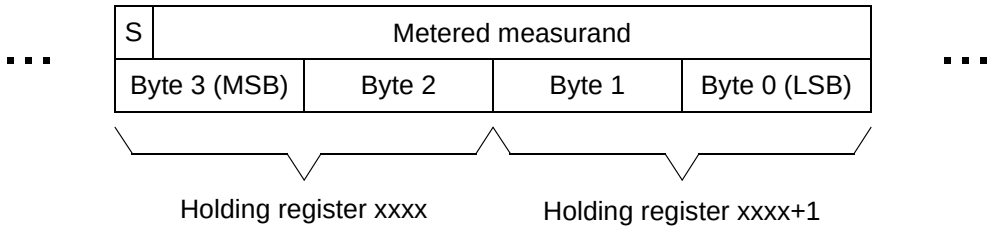


Bild 3-4 Data type: Metered measurand (unsigned long)

Status bit (S) The status bit is set for invalid metered measurands in case of:

- corruption of the metered measurand after device reset/device start-up (the status bit is deleted after two update cycles of the metered measurand after device reset/device start-up),
- the external error bit of a pulse counter (metered measurand with a pulsed binary input as source) is set.

Note:

The overflow of the metered measurands is $7FFFFFFF_{hex}$ + 1 and the counter then starts at 0 again.

3.5 Transformer tap change command (TC)

Range of values

1 - LOWER

2 - RAISE

Hinweis:

- The transmission of transformer tap change commands is carried out via one holding register per transformer tap change command.
- Only the two above indicated values for LOWER and RAISE are evaluated at write accesses on this Holding register.
  A write access with another value is rejected in the SIPROTEC device.

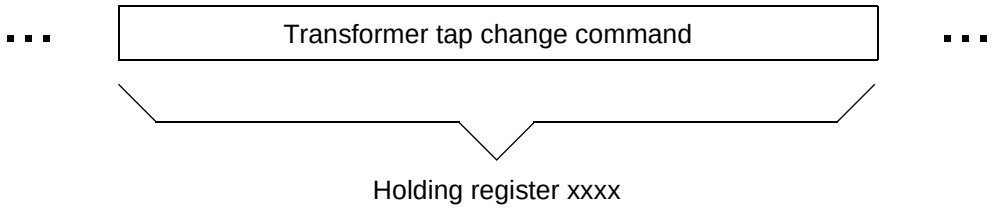


Bild 3-5 Data type: Transformer tap change command

3.6 Transformer tap position indication (TM)

Range of values 1 to 62

(63 = $3F_{hex}$ = invalid transformer tap position value via binary inputs)

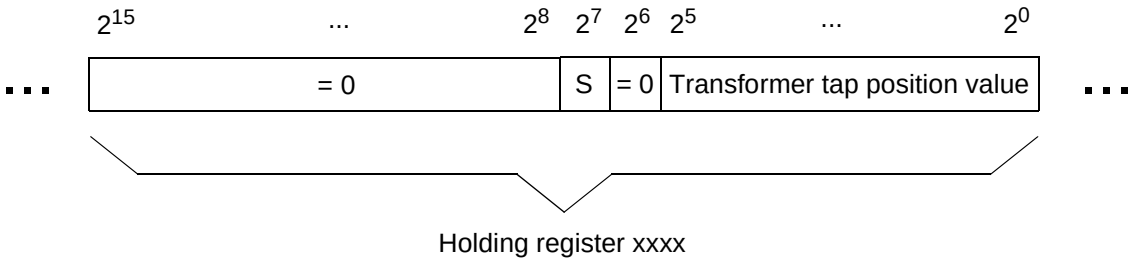


Bild 3-6 Data type: Transformer tap position indication

Status bit (S) In the event of a set status bit the transformer tap position indication is invalid, as no relevant binary inputs are allocated in DIGSI.
In conjunction a transformer tap position value of 0 is transmitted.

3.7 Time/Date

The Time/Date format is used for:

- Time synchronization of the SIPROTEC device via Modbus (ref to chap. 4.5),
- Recording-time of peak or minimum demand values of a measured value.

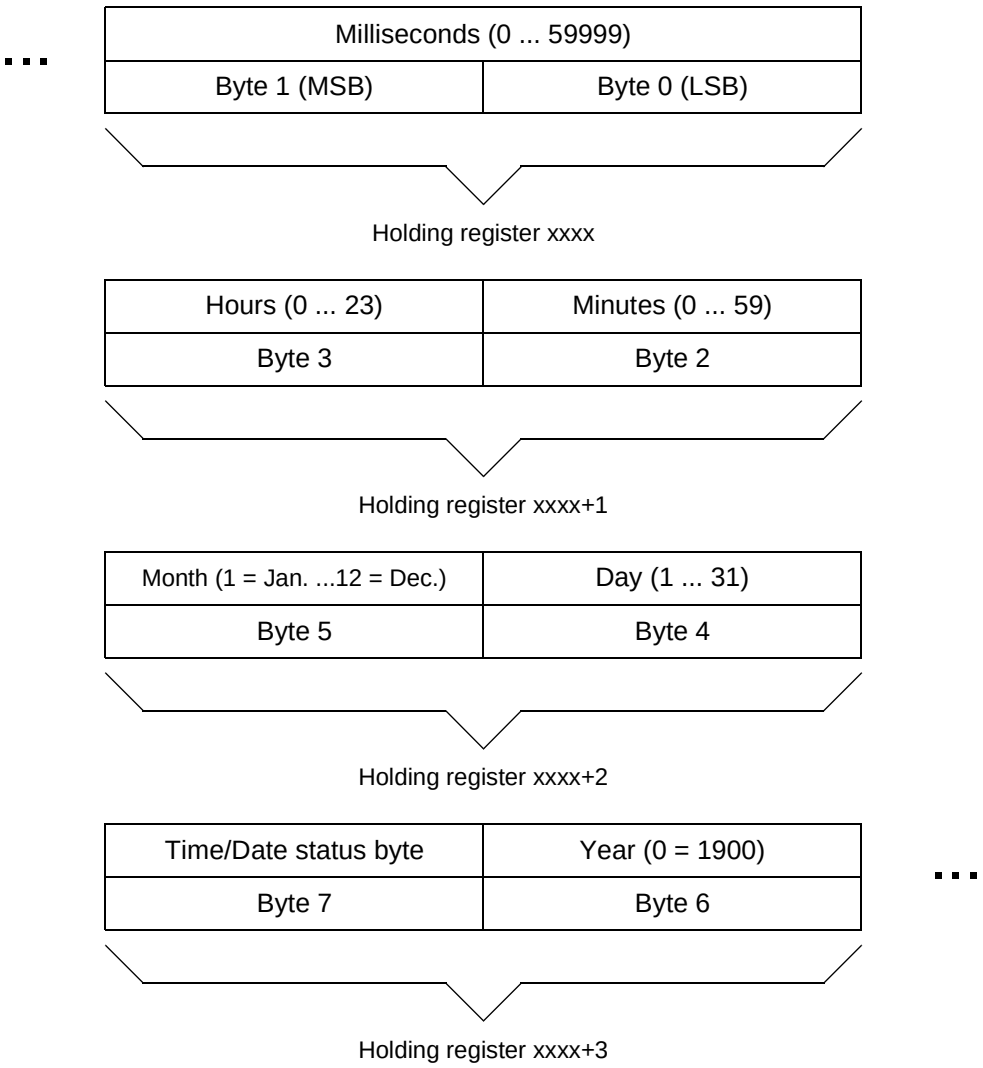


Bild 3-7 Data type: Time/Date

Time/Date status OR logical operation of the following status bits in the Time/Date status byte:

$10_{hex} \rightarrow$ Daylight savings time

$20_{hex} \rightarrow$ Time/Date failure

$40_{hex} \rightarrow$ Time/Date invalid

3.8 Message block for Event recorder

The complex data type “Message block” defines an entry in the Event recorder.

Further information regarding the properties as well as the retrieval methods of the Event recorder can be found in chapter 2.5.

Holding register xxxx	Register type	Bit offset
	Byte 1	Byte 2
Holding register xxxx+1	Register address	
	Byte 3 (MSB)	Byte 4 (LSB)
Holding register xxxx+2	Message cause	Indication type
	Byte 5	Byte 6
Holding register xxxx+3	Value	
	Byte 7 (MSB)	Byte 8 (LSB)
Holding register xxxx+4	Milliseconds (0 ... 59999)	
	Byte 9 (MSB)	Byte 10 (LSB)
Holding register xxxx+5	Hours (0 ... 23)	Minutes (0 ... 59)
	Byte 11	Byte 12
Holding register xxxx+6	Month (1 = Jan. ...12 = Dec.)	Day (1 ... 31)
	Byte 13	Byte 14
Holding Register xxxx+7	Clock status	Year (0 = 1900)
	Byte 15	Byte 16

Bild 3-8 Data type: Message block for Event recorder

Bytes 1 to 4: Identification

The first four bytes identify an annunciation and correspond to the parameter "Register type", "Register address" and "Mask" selected in DIGSI (ref. to chap. 4.3).

Byte 1: Register type

Modbus register type, supported values:

0 = Coil Status register

1 = Input Status register

4 = Holding register

Byte 1 directly corresponds to the parameter "Register type" in DIGSI.

The byte "Register type" has the value FF_{hex} if the Message block does not contain valid data.

Byte 2: Bit offset

This value is only relevant, if a single-point indication or double-point indication is to evaluate (ref. to Byte 6) and the "Register type" is Holding register (Byte 1 = 4).

The Bit offset indicates the bit position (0 to 15) in the Holding register, the position of the least significant bit if double-point indication.

Byte 2 can be derived from parameter "Mask" if "Register type" = 4 in DIGSI:

Mask = 1 → Bit offset = 0; Mask = 2 → Bit offset = 1; Mask = 4 → Bit offset = 2;

Mask = 8 → Bit offset = 3; Mask = 16 → Bit offset = 4; etc.

Bytes 3 and 4: Register address

These two bytes indicate the register address of the annunciation.

This corresponds to the register address according to the Modbus specification (equals register number - 1) with a value range of 0 to 65535.

Example

The single-point indication "Error sum alarm" (ref. to Figure 4-8) is for 7SJ6x devices:

- Preallocated at Register 40129, Bit position 2^4 according to the bus mapping documentation and
- routed to "Register type" = 4, "Register address" = 128, "Mask" = 16 in DIGSI.

As identification in the Message block is entered:

Register type = 4, Bit offset = 4, Register address = 128

Byte 5: Message cause

It can be parameterized, if message causes are entered in the Event recorder entries or not (ref. to chap. 2.1.2).

If message causes are not used:

- Byte 5 always has the value 0,
- only annunciations with actual values (SIPROTEC message causes SPN, INT, FB+) are transmitted.

Byte 5	Message cause	Corresponds to message cause in the operational event log of the SIPROTEC device	Annunciation value corresponds to
0	Spontaneous	SPN	Actual value
1	Intermediate position	INT	Actual value
2	Control issued or Tagging positive	CO+, TG+	Must value
3	Control abortion (from local control)	CA+	Must value
4	Control rejected or Tagging negative	CO-, TG-	Must value
5	Feedback negative	FB-	Must value
6	Feedback positive	FB+	Actual value

Tabelle 3-1 Message block: Message causes

Byte 6: Indication type

Byte 6 describes the type of the information and contains bits for indicating annunciations which are part of a General scan cycle.

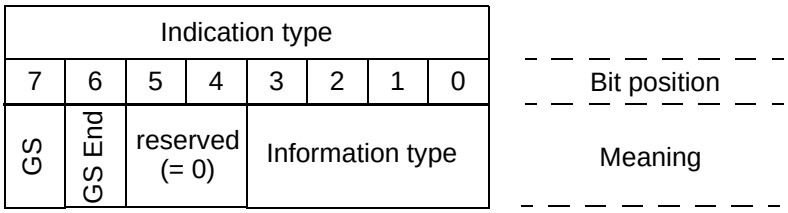


Bild 3-9 Message block: Indication type

Bits 0 to 3: Information type

$0001_{bin} = 1_{hex}$ for single-point indications

$0010_{bin} = 2_{hex}$ for double-point indications

$0011_{bin} = 3_{hex}$ for transformer tap position indications

$0100_{bin} = 4_{hex}$ for statistic values and fault currents

Bit 7: General scan bit (GS)

A General scan can be initiated by:

- Start-up General Scan:
  Following initial or reboot, all start-up values of the objects which are parameterized for recording are entered in the Event recorder (this option is switched off by default, ref. to chap. 2.1.2).
- General scan request from Modbus master:
  The Modbus master can trigger a General scan cycle by setting/writing a bit (“Start GS”) in the “SOE_Control” register (ref. to chap. 2.5.2.3).
  All objects which are parameterized for recording are entered in the Event recorder.

The General scan bit is set for every entry in the Event recorder which causes from a General scan.

During a General scan cycle, the values of the annunciations are explicitly (sequentially) requested from the SIPROTEC device and have the time stamp of the response from SIPROTEC device.

Bit 6: End of General scan (GS End)

This bit is set (in addition to bit 7 “General scan bit”) in the entry of the last annunciation of a General scan cycle and indicates the end of this General scan cycle.

Byte 7 and 8: Value

These bytes contain the value of the annunciation after the change.

Single-point indications:
1 = ON / COMING, 0 = OFF / GOING

Double-point indications:
0 = “Not applicable” or Intermediate position (ref. to chap. 3.2)
2 = ON, 1 = OFF, 3 = Intermediate position

Transformer tap position indications:
1 to 62 = transformer tap position value,
63 = invalid transformer tap position value,
128 = no binary input is routed to the transformer tap position indication

Statistic values and fault currents:
signed 16-bit value; 8000_{hex} if invalid or overflow

Bytes 9 to 16: Time stamp

Time stamp (milliseconds to year) at the time of the change of the annunciation.

Meaning of the Clock status byte:

OR logical operation of the following status bits

$10_{hex} \rightarrow$ Daylight savings time

$20_{hex} \rightarrow$ Time/Date failure

$40_{hex} \rightarrow$ Time/Date invalid

Modbus - Parameterization in DIGSI

4

This chapter describes the parameterization of Modbus as the system interface of a SIPROTEC device using parameterization system DIGSI.

For device parameterization have to be used:

- DIGSI 4.21 considering the preconditions explained in chapter 4.1.3,
- DIGSI 4.3 or higher,
- DIGSI 4.4 or higher at use of
  - Transformer tap change commands/Transformer tap position indications (ref. to chap. 3.5 and 3.6).

4.1 Interface selection and mapping files

Precondition

The parameterization of Modbus for a SIPROTEC device requires:

- selection of Modbus as system interface,
- selection of a mapping file which fixes the allocation of the data objects of the SIPROTEC device to the positions in the Modbus messages.

Bus specific parameters have to be defined simultaneously when selecting a mapping file (ref. to chap. 2.1).

4.1.1 Standard mappings 3-1 to 3-n

A number of standard mappings (standard mapping 3-1 to standard mapping 3-n, n = device type dependent number of standard mappings) are available for every SIPROTEC device type.

These mappings are different in the data size available via Modbus and offer a standard pre-allocation of SIPROTEC data objects to Modbus register positions.

Adaption of the allocation

In adaptation to the concrete installation environment the standard allocation can be changed (ref. to chap. 4.3):

- removing of data objects from Modbus register positions,
- routing of data objects to free Modbus register positions,
- scaling of measured values according to the operating values of the primary equipment.

Note:

The data size available via Modbus (number of commands, annunciations, measured values, metered measurands) is exclusively fixed by the selection of a standard mapping.

The data in the Modbus registers are defined in the bus mapping documentations of the individual SIPROTEC devices (ref. to page 3).

4.1.2 Compatibility with standard mappings of previous versions

Standard mappings 1 to n

The standard mappings 1 to standard mapping n (n = device type dependent number of standard mappings) enclosed to DIGSI 4.1 for SIPROTEC devices 7SJ61...7SJ64 and 6MD63 should not be used for new device parameterizations.

A customization of allocations and scalings is not possible with these mappings and they have functional limitations as compared with the descriptions in this manual.

4.1.3 Interface selection and mapping selection in DIGSI 4.21

Attention!

The parameterization of Modbus with the functionalities described in this manual using standard mapping 3-1 to 3-n (n = device type dependent number of standard mappings) and DIGSI 4.21 requires the following preconditions:

- Update of the Parameter generating DLL for communication modules (“PG.DLL”) in the DIGSI directory on the PC to version V02.04.01 or higher.

The update is required because of the extended functionalities of the standard mappings 3-1 to 3-n.
If no update of the Parameter generating DLL is carried out then an error occurs during the translation process of the mapping file when closing the dialog window **`Properties - SIPROTEC 4 device`** (ref. to Figure 4-1).

You can download the Parameter generating DLL V02.04.01 as

DIGSI 4.21 ServicePack 3

from Internet **http://www.siprotec.com**

or please contact your Siemens representative.

When entering the device MLFB (order number) to create a new SIPROTEC device in the **DIGSI 4 Manager** for parameterization, you are asked automatically for the selection of Modbus as system interface if the SIPROTEC device has a Modbus communication module at delivery.

Changing the system interface to Modbus for already existing devices in DIGSI is also possible.

Protocol assignment for system interface

Select the SIPROTEC device in your project in the **DIGSI 4 Manager** and use the menu item ***Edit - Object properties...*** to open the **Properties - SIPROTEC 4 device** dialog window (ref. to Figure 4-1).

In the property sheet **Communications Modules** the entry “additional protocols, see MLFB Ext. L” has to be selected for “11. SYSTEM-Port”.

By pressing the button “L: ...” the dialog window **Additional information** is opened which is used to enter the type of the communication module.

Please select in the dialog window **Additional information**:

- “Protocol” or “none” (depending on the SIPROTEC device type) for “1. SYSTEM-Port” and
- “Modbus, RS485” or “Modbus, 820nm fiber, ST-Connector” (depending on the hardware composition of the SIPROTEC device) for “2. SYSTEM-Port”.

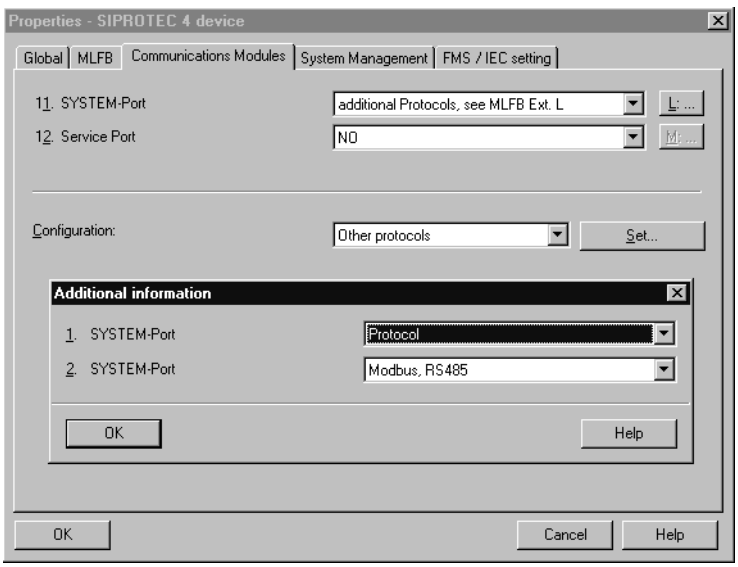


Figure 4-1 DIGSI 4.21: Modbus protocol assignment

Mapping file

The mapping file selection is available in the dialog window **Other protocols** which is opened by pressing the button “Set ...” in the property sheet **Communications Modules** (ref. to Figure 4-1).

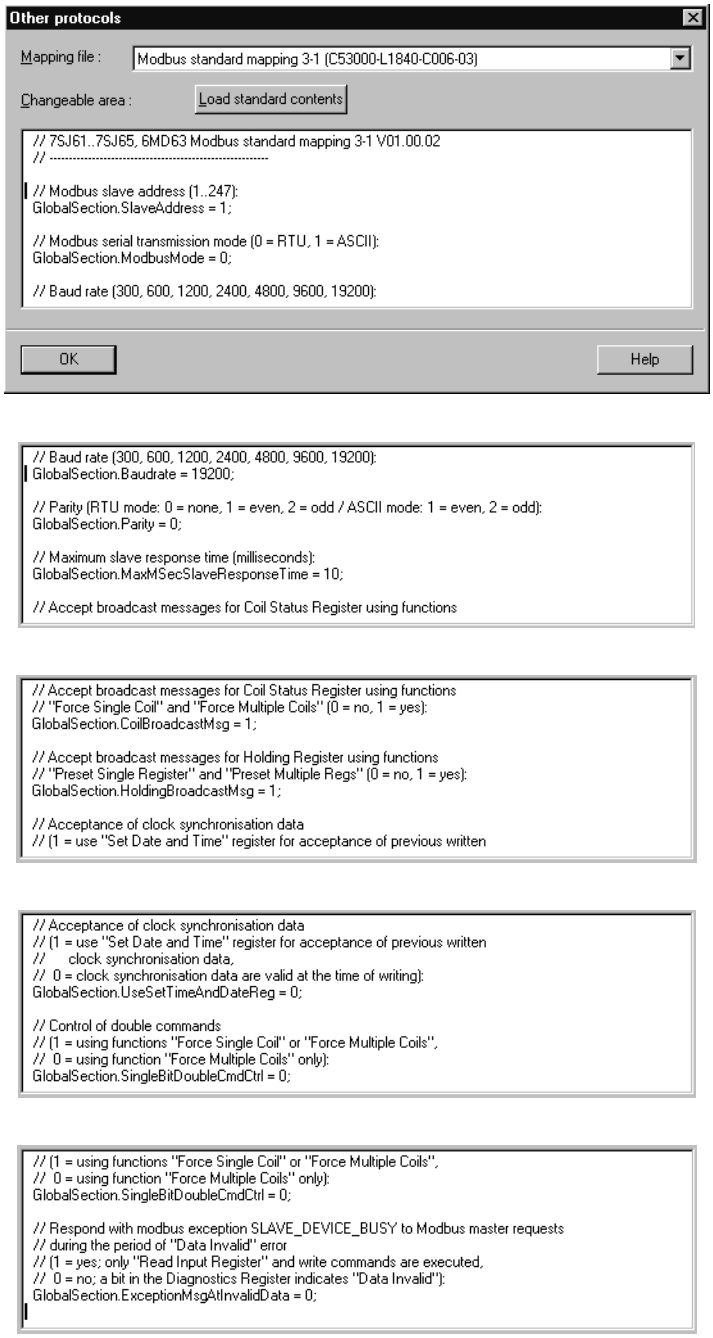


Figure 4-2 DIGSI 4.21: Mapping file selection and bus specific parameters

The list box “Mapping file:” includes all available Modbus mapping files for the respective SIPROTEC device type with their name and a reference to the associated bus mapping documentation (order numbers, ref. to page 3).

In the edit area “Changeable area:” bus specific parameters can be changed. Please refer to chap. 2.1 for a description of these parameters.

The button “Load standard contents” restores the default values of the bus specific parameters.

Note:

Please, edit only the numbers in the rows which do not start with “//” and note the semicolons at the end of the line.

Editing anything else in the “Changeable area:” may cause an error when closing the **`Properties - SIPROTEC 4`** device dialog window.

Attention!

If after change of a measured value’s scaling (ref. to chap. 4.4) a bus specific parameter is changed then all scalings are reset to their defaults according to the bus mapping documents (ref. to page 3) again.

4.1.4 Interface selection and mapping selection in DIGSI 4.3 or higher

When entering the device MLFB (order number) to create a new SIPROTEC device in the **DIGSI 4 Manager** for parameterization, you are asked automatically for the selection of Modbus as system interface if the SIPROTEC device has a Modbus communication module at delivery.

Changing the system interface to Modbus for already existing devices in DIGSI is also possible.

Protocol assignment for system interface

Select the SIPROTEC device in your project in the **DIGSI 4 Manager** and use the menu item ***Edit - Object properties...*** to open the **Properties - SIPROTEC 4 device** dialog window (ref. to Figure 4-3).

In the property sheet **Communications Modules** the entry “additional protocols, see MLFB Ext. L” has to be selected for “11. SYSTEM-Port”.

By pressing the button “L: ...” the dialog window **Additional information** is opened which is used to enter the type of the communication module.

Please select in the dialog window **Additional information**:

- “Protocol” or “none” (depending on the SIPROTEC device type) for “1. SYSTEM-Port” and
- “Modbus, RS485” or “Modbus, 820nm fiber, ST-Connector” (depending on the hardware composition of the SIPROTEC device) for “2. SYSTEM-Port”.

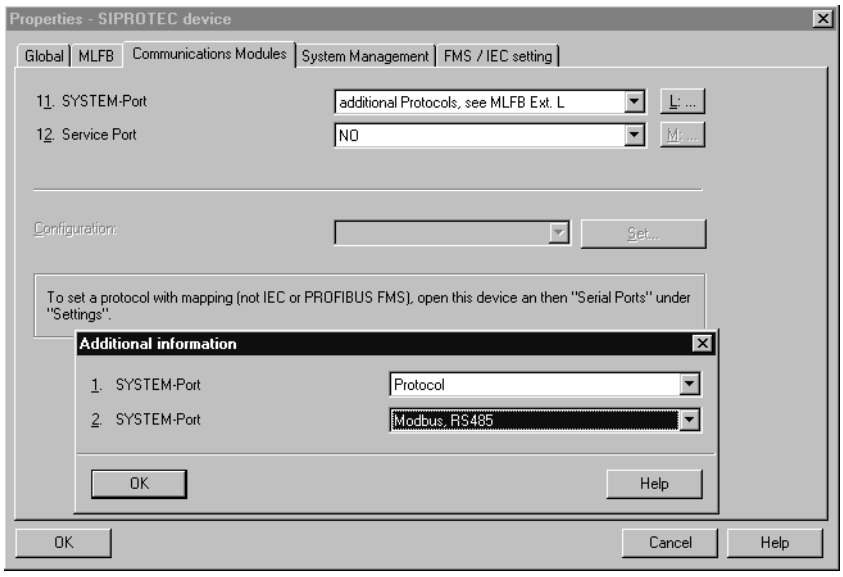


Figure 4-3 DIGSI 4.3: Modbus protocol assignment

Mapping file

To select a mapping file, please open the SIPROTEC device in DIGSI.

The dialog window **Interface Settings** (in DIGSI via ***Settings*** - ***Serial Ports***) offers in the property sheet **Supplementary protocols at device** the following dialog elements:

- display of the chosen communication module
  (ref. to page 4-61, “Protocol assignment for system interface”),
- the list box “Mapping file:” which includes all available Modbus mapping files for the respective SIPROTEC device with their name and a reference to the associated bus mapping documentation (order numbers, ref. to page 3),
- the edit area “Module-specific settings:” to change the bus specific parameters (ref. to chap. 2.1 for a description of these parameters).

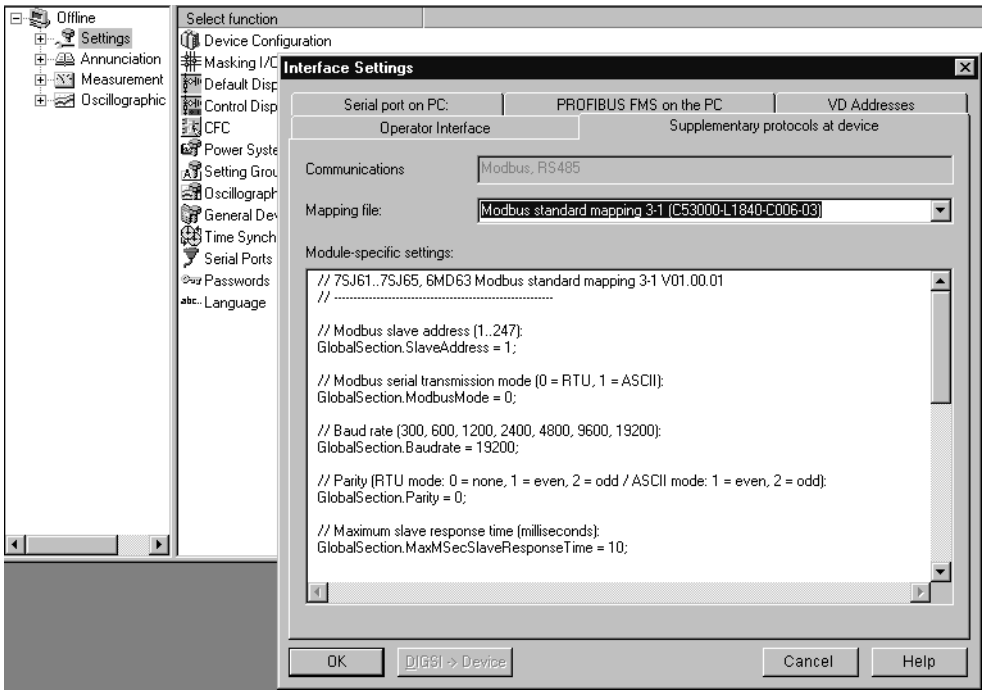


Figure 4-4 DIGSI 4.3: Mapping file selection and bus specific parameters

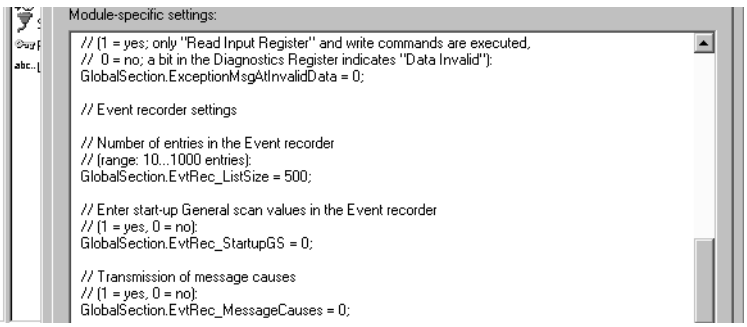


Figure 4-5 DIGSI 4.3: Bus specific parameters for Event recorder

List box “Mapping file”

If no mapping file is currently assigned to the SIPROTEC device then the following entries are available in the list box “Mapping file”:

Selection	Meaning
<none>	No mapping file is assigned to the device.
Modbus standard mapping 3-1 ... Modbus standard mapping 3-n	Selection of a mapping file 3-1 to 3-n (n = device type dependent number of standard mappings).

The following entries can be selected at an already existing mapping file assignment:

Selection	Meaning
<none>	No mapping file is assigned to the device.
<see module-specific settings>	This selection indicates the currently to the device assigned mapping file with the changes of bus specific parameters already carried out in the edit area “Module-specific settings”. Number and version of the mapping file have to be taken from the first line in the edit area "Module-specific settings".
Modbus standard mapping 3-1 ... Modbus standard mapping 3-n	(New) Selection of a mapping file 3-1 to 3-n (n = device type dependent number of standard mappings). All bus specific parameters are reset to default values.

If the mapping file assignment was changed for a SIPROTEC device, then this is in general connected with a change of the routing of the SIPROTEC objects to the system interface.

Please, check after choice of a new mapping file the allocations to "Destination system interface" or "Source system interface" in the **`DIGSI configuration matrix`**.

Edit area “Module-specific settings”

Please, edit only the numbers in the rows which do not start with “//” and note the semicolons at the end of the line.

Editing anything else in the “Module-specific settings” may cause an error when closing the **`Interface Settings`** device dialog window.

4.2 Numbering of Modbus registers

4.2.1 Modbus specification

Generally, for numbering of Modbus registers it is to distinguish between:

- the register number and
- the register address.

Register number

The register number is used to identify a Modbus register, normally with a five-digit decimal number in which the highest-order digit defines the register type:

- 0 - Coil Status register
- 1 - Input Status register
- 3 - Input register
- 4 - Holding register

The count of register numbers starts at 1 per register type, e.g.:

- 00127 = Coil Status register 127 (alternative: Coil 127),
- 40108 = Holding register 108.

Register address

The register address is used for address information in Modbus messages and is related to the value 0:

For this reason the following relation between register number and register address exists:

Register address = Register number - 1

For the above example follows:

- Coil 127 is addressed as 126 ($007E_{hex}$) in a Modbus message for reading or writing the Coil Status register,
- 40108 is addressed as 107 ($006B_{hex}$) in a Modbus message for reading or writing the Holding register.

4.2.2 Bus mapping documentations of the SIPROTEC devices

In the Modbus bus mapping documentations of the individual SIPROTEC devices (ref. to page 3) the **register number** is used for identification of an information in the Modbus register map.

4.2.3 Parameterization in DIGSI

Coil Status register, Input Status register

The Coil Status registers and Input Status registers are organized in the Modbus slave of the SIPROTEC device in groups of 16 bits.

For this reason the identification of such a register in DIGSI requires the selection of:

- a **base register address** (a value divisible by 16) and
- a **mask** which describes the position in the group by a 1-bit.

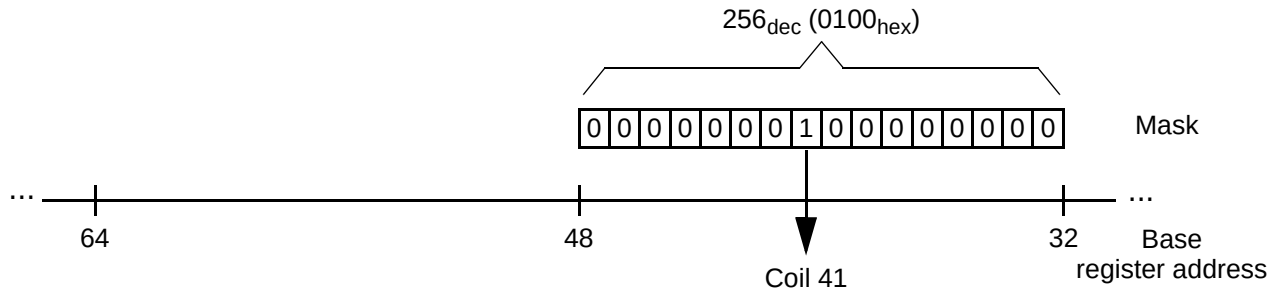


Figure 4-6 Base register address and mask for Coil Status and Input Status registers

Example (see above):

The Coil Status register 41 (register address 40 in a Modbus message) is addressed in DISGI as follow:

- Base register address = 32,
- Mask = 256_{dec} (0100_{hex})

Input register

Register addresses are used for the definition of the position of an information in the Input registers.

Data objects in the Input registers always have a 16 bits wide data type, i.e. a value occupies one Input register.

Holding register

Register addresses are used for the definition of the position of an information in the Holding registers.

In addition the following rules are valid:

- The storage of variables of more complex data types in the Modbus Holding registers (i.e. variables greater than one holding register, e.g. metered measurands) is processed according to the following convention:

  The register which has the lower address contains the most significant byte (MSB) of the variable and the register with the higher address contains the least significant byte (LSB).
  The parameterized register address for this information indicates the position of the most significant byte.

- If a Holding register is used to store bit-information (e.g. 16 single-point indications), then additionally to the register address a **mask** with a 1-bit at the position of the bit-information in the Holding register has to be defined.

4.3 Customization of the allocations

The identification whether an information is routed on system interface (Modbus) is shown in the columns "Source system interface" and "Destination system interface" in the **DISGI Configuration matrix**.

A cross ('X') in this column indicates the associated information as "routed on system interface".

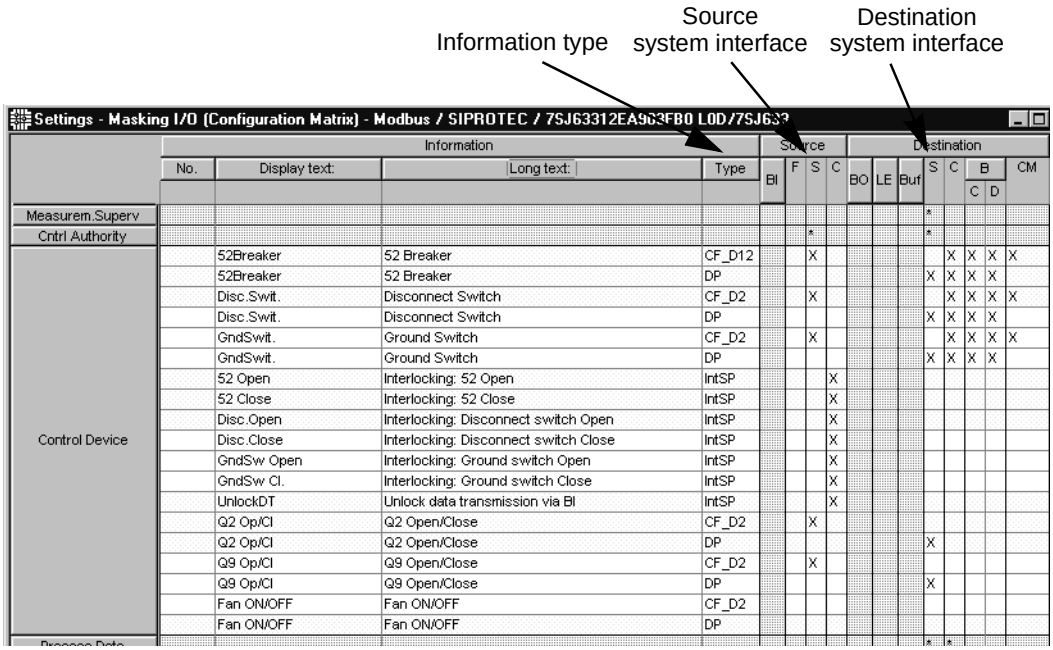


Bild 4-7 DIGSI Configuration matrix with columns for system interface routing

Source system interface

The SIPROTEC object can be controlled via Modbus.

This is possible for the following information types:

IntSP	Internal single-point indication (tagging)
IntDP	Internal double-point indication (tagging)
SC/DC	Single control/Double control without feedback from process
SF/DF	Single control/Double control with feedback from process

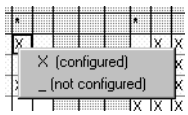
Destination system interface

The value of the SIPROTEC object is transmitted to the Modbus master.

This is possible for the following information types:

SP, DP	Single-point indication, Double-point indication
Out	Output annunciation
IntSP	Internal single-point indication (tagging)
IntDP	Internal double-point indication (tagging)
MV	Measured value
MVMV	Power meter (metered measurand, source is a measured value)
PMV	Pulse (metered measurand, source is a pulsed binary input)
TM	Transformer tap position indication

To add or remove an information to "Source system interface" or "Destination system interface" set/reset the cross ('X') in the associated column of the **DIGSI Configuration matrix** (pop-up menu when pressing the right mouse button).



Note:

- The max. number of routable objects of an information type varies according to the chosen mapping file.
  If e.g. a measured value not routed in the mapping file per default shall be transferred via Modbus, then first a measured value already routed has to be removed from system interface so that the position gets available in a Modbus register.
- An error message is shown if all routing possibilities of an information type are occupied and if it is nevertheless tried to route an information of this type.

Adding an allocation

Adding an allocation requires (in addition to the identification in the system interface column of the **DIGSI Configuration matrix**) the selection of the position of the information in the Modbus registers as well as the definition of scaling values for measured values (scaling of measured values ref. to chap. 4.4).

Therefore after adding the allocation, the **Object properties** dialog window which is used to define the position of the information is opened automatically (property sheet **Protocol info-Source** or **Protocol info-Destination**).

Using Modbus mappings without Event recorder option:

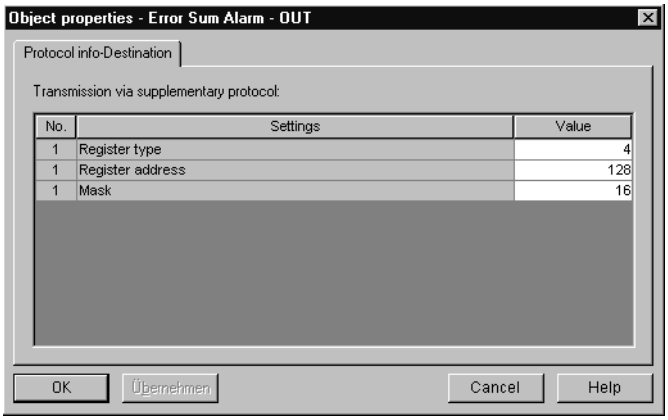


Figure 4-8 Definition of the position of an information

If the Modbus mapping of the SIPROTEC device supports an Event recorder (ref. to chap. 2.5) then it must be selected whether an annunciation is to be entered in the Event recorder if changed or not.

Therefore, an additional parameter “Add to Event recorder” has to be set in the **`Object properties`** dialog window:

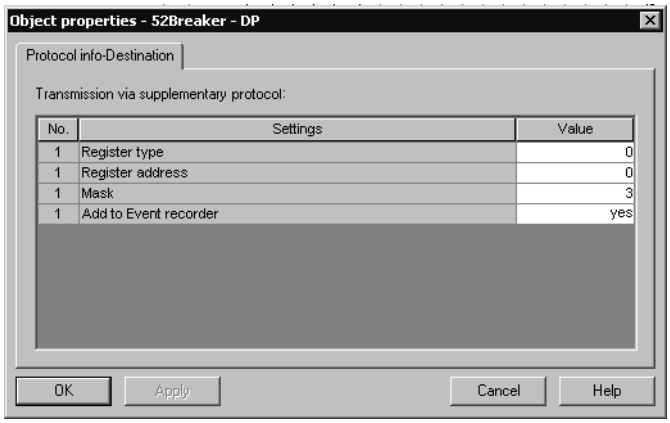


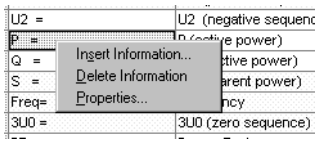
Figure 4-9 Adding an annunciation to the Event recorder

Attention!

- The information type of double commands for routing as "Source system interface" depends on the information type of the associated checkback indication.
  Only double commands with a double-point indication as checkback indication can be routed to the positions in the Modbus registers which are reserved for double commands according to the bus mapping.
  Double commands with a single-point indication as checkback indication or without checkback indication are treated via Modbus like single commands and must be routed to the positions reserved for single commands.
- A command and its associated checkback indication have to be routed to the same positions in the Coil Status register block (commands as “Source system interface”, checkback indications as “Destination system interface”).
- A transformer tap change command and its associated checkback indication (transformer tap position indication) have to be routed to the same positions in the Holding register block (commands as “Source system interface”, checkback indications as “Destination system interface”).
- Commands without checkback indication can not be routed to “Destination system interface”. Reading the state of these command types is not possible via Modbus.

Change of an existing allocation

If an information already routed on system interface shall get another (empty) Modbus register position, then the **Object properties** dialog window (ref. to Figure 4-8) has to be selected (***Properties ...*** in the pop-up menu when pressing the right mouse button in one of the columns "Display text", "Long text" or "Type" of the **DIGSI Configuration matrix** in the row which is associated to the information).



Depending on the information type the following parameters are to select in the property sheet **Protocol info-Source** or **Protocol info-Destination** of the **Object properties** dialog window:

Parameter	Comments	Info types for Protocol info source	Info types for Protocol info destination
Register type (ref. to chap. 4.2.1)	Register type (0, 1, 3, 4)	all	all
Register address (ref. to chap. 4.2.3)	Base register address for Coil Status registers and Input Status registers	IntSP, IntDP, SC/DC, SF/DF	SP, DP, Out, IntSP, IntDP
	Register address for Input registers and Holding registers	SC/DC, SF/DF (for transformer tap change commands)	SP, Out, MV, IntSP, MVMV, PMV, TM
Mask (ref. to chap. 4.2.3)	Position in the Coil Status registers or Input Status registers regarding to the base register address	IntSP, IntDP, SC/DC, SF/DF	SP, DP, Out, IntSP, IntDP
	Position of bit-information in a Holding register	-	SP, Out, IntSP

Tabelle 4-1 Parameters for Protocol info-Source and Protocol-info Destination

Example

The information “Error sum alarm” (ref. to Figure 4-8) is transferred after routing to “Destination system interface” in the register number 40129 (Holding register, register address = 128) at bit position 2^4 ($16_{dec} = 0010_{hex} = 0000000000010000_{bin}$).

Note:

Only the positions in the Modbus registers are offered to the selection on which the information type still can be routed according to the mapping file and the current occupancy.

Behavior at not routed positions in the Modbus registers:

- In input direction the Modbus master always reads the value 0 from these positions.
- A write command to not routed positions in the Coil Status registers and Holding registers is ignored by the SIPROTEC device.

Binary incoming annunciations

Binary incoming annunciations (marked with the sign '>' in the name, e.g. ">BLOCK 81-1") cannot be routed directly as "Source system interface".

A control of these objects via Modbus as a substitute for using binary inputs is however often meaningful.

To do this, taggings (information type: IntSP) routed to "Source system interface" as well as "Destination CFC" are used.

The binary incoming annunciation which is routed as "Source CFC" is connected via a CONNECT module in CFC to the tagging.

Example

Control of object ">BLOCK 81-1" using a user-defined tagging "DP 81-1 bl" via Modbus:

- In the **DIGSI Configuration matrix** set the source for ">BLOCK 81-1" to CFC output.
- Create a user-defined tagging using the **Information catalog**, rename this object in “DP 81-1 bl”, release it as “Destination CFC” and route it to an empty position in the Modbus Coil Status registers as “Source system interface”.
- Open a **CFC working page** (e.g. “Device, Systemlogic” in run sequence PLC1).
- Insert a CONNECT module and check the run sequence of this module (PLC1_BEA, MW_BEARB, ...) according to the selected **CFC working page**.
- Connect the input ("BO X") of the CONNECT module with the operand "DP 81-1 bl".
- Connect the output ("Y BO") of the CONNECT module with the operand ">BLOCK 81-1".
- Save and translate the **CFC working page**.

The object ">BLOCK 81-1" (and with that the associated protective function) can be influenced by changing the value of the "DP 81-1 bl" via Modbus now.

4.4 Scaling of measured values

Measured values are transferred via Modbus between the SIPROTEC device and the Modbus master as integer values (two bytes, ref. to chap. 3.3) but they are in general available in the SIPROTEC device in floating-point format as a percentage referred to the parameterized nominal values of the primary equipment.

4.4.1 Measurement conversion

Before transmission of a measured value via Modbus a measurement conversion (scaling) must be carried out in the SIPROTEC device.

Scaling

Scaling of a measured value to the format for the transmission via Modbus means the definition of:

- Type,
- Scaling factor,
- Zero offset.

Type

Decision, whether the measured value is transmitted via Modbus as percentage value, primary value or secondary value.

Depending on the measured value not all of these three possibilities are available, e.g. no secondary values for power values.

Scaling factor

The measured value in the SIPROTEC device (floating-point format) is multiplied by the scaling factor before transformation to an integer value (for Modbus).

It is possible to transfer fractional digits by multiplication by a multiple of 10 in the integer value with that.

Zero offset

The Zero offset is added to the result of the multiplication of the measured value in the SIPROTEC device (floating-point format) by the scaling factor.

Formula

The measured value in integer format for transmission via Modbus is calculated summarizing according to the following formula:

$$\text{Measured value}_{\text{Integer}} = \text{Measured value}_{\text{Float}} * \text{Scaling factor} + \text{Zero offset}$$

in which "Measured $\text{value}_{\text{Float}}$" is a percentage value or, if necessary, changed into primary value or secondary value before (according to the definition of Type).

4.4.2 Number representation depending on the parameterization

For specification of the scaling of a measured value it must be known in which number format (number of the relevant fractional digits) the measured value is available in the SIPROTEC device and to which unit it refers.

Percentage value

A scaling factor of 100 is recommended for percentage values.

With that the "Measured value$_{Integer}$" in the Modbus message has to be interpreted as:

+/- 32767 corresponds to +/- 327.67 %

Secondary value

The transmission of a measured value as secondary value is meaningful only in few cases (e.g. transducer measured values in mA).

The number of significant fractional digits depends on the installations and transducers data.

Primary value

The number of relevant fractional digits and the respective unit for primary values depends tightly on the parameterized nominal values of the primary equipment (DIGSI: **`Power system data 1`** and **`Power system data 2`**).

Note:

The following parameter numbers apply to the devices 7SJ61...7SJ64 and are partly divergently for other device types.

`Voltages:` V_a, V_b, V_c, V_{a-b}, V_{b-c}, V_{a-c}, $3V_0$, V1, V2

Parameter: 1101 Nominal operation voltage of primary equipment

Parameter area	Number representation / unit
1.00 ... 10.00 kV	0.00 ... 99.99 kV
10.01 ... 100.00 kV	0.0 ... 999.9 kV
100.01 ... 1000.00 kV	0 ... 9999 kV
greater than 1 MV	0 ... 99.99 MV

`Displacement voltages:` V_N

Parameter: 0202 Voltage transducer - Primary voltage
0206 Ratio factor V_{ph}/V_{delta}

Product of parameters 0202 and 0206	Number representation / unit
100.00 ... 1000.00 V	0 ... 9999 V
1.01 ... 10.00 kV	0.00 ... 99.99 kV
10.01 ... 100.00 kV	0.0 ... 999.9 kV
100.01 ... 1000.00 kV	0 ... 9999 kV
greater than 1 MV	0.00 ... 99.99 MV

Currents: I_a, I_b, I_c, $3I_0$, I1, I2

Parameter: 1102 Nominal operating current of primary equipment

Parameter area	Number representation / unit
10.00 ... 100.00 A	0.0 ... 999.9 A
100.01 ... 1000.00 A	0 ... 9999 A
1.01 ... 10.00 kA	0.00 ... 99.99 kA

Ground currents: I_N, I_{Ns}

Parameter:

(V4.2) 0204 Current transducer - Rated primary current
0207 Ratio factor I_N / I_{ph} or (dependent on the device type)
0208 Ration factor I_{Ns} / I_{ph}
(V4.3) 0217 Rated primary current of I_N Current transducer

Product of parameters 0204 and 0207 / 0204 and 0208 or parameter 0217	Number representation / unit
0.00 ... 1.00 A	0 ... 9999 mA
1.01 ... 10.00 A	0.00 ... 99.99 A
10.01 ... 100.00 A	0.0 ... 999.9 A
100.01 ... 1000.00 A	0 ... 9999 A
1.01 kA ... 10.00 kA	0.00 ... 99.99 kA
greater than 10 kA	0.0 ... 999.9 kA

Power: P, Q, S

Parameter: 1101 Nominal operation voltage of primary equipment
1102 Nominal operating current of primary equipment

Product of parameter 1101 and 1102 multiplied by $\sqrt{3}$	Number representation / unit
10.00 ... 100.00 kW (kVAR)	0.0 ... 999.9 kW (kVAR)
100.01 ... 1000.00 kW (kVAR)	0 ... 9999 kW (kVAR)
1.01 ... 10.00 MW (MVAR)	0.00 ... 99.99 MW (MVAR)
10.01 ... 100.00 MW (MVAR)	0.0 ... 999.9 MW (MVAR)
100.01 ... 1000.00 MW (MVAR)	0 ... 9999 MW (MVAR)
1.01 ... 10.00 GW (GVAR)	0.00 ... 99.99 GW (GVAR)
greater than 10 GW (GVAR)	0.0 ... 999.9 GW (GVAR)

Example **Definition of the scaling for a power measurement value**

In the parameter set is configured:

Nominal operation voltage of primary equipment = 12.00 kV

Nominal operating current of primary equipment = 100 A

It follows:

$V_{nom} * I_{nom} * \sqrt{3}$ = 2078.46 kW = 2.078 MW

In the SIPROTEC device the power measurement value is available with the following number representation and unit (see table above), i.e. two relevant fractional digits:

0.00 ... 99.99 MW

According to this a scaling factor of 100 is meaningful for transmission of the information as an integer value.
If the scaling factor is less than 100 then relevant fractional digits are lost.
A scaling factor of greater than 100 does not give more precision because there are only two relevant fractional digits.

With that the "Measured $value_{Integer}$" in the Modbus message has to be interpreted as:

+/- 32768 corresponds to +/- 327.68 MW

4.4.3 Parameterization in DIGSI

The **Object properties** dialog window (ref. to chap. 4.3) contains for measured values - beside the property sheet **Protocol info-Source** or **Protocol info-Destination** - an additional property sheet titled **Measured value-Destination**.

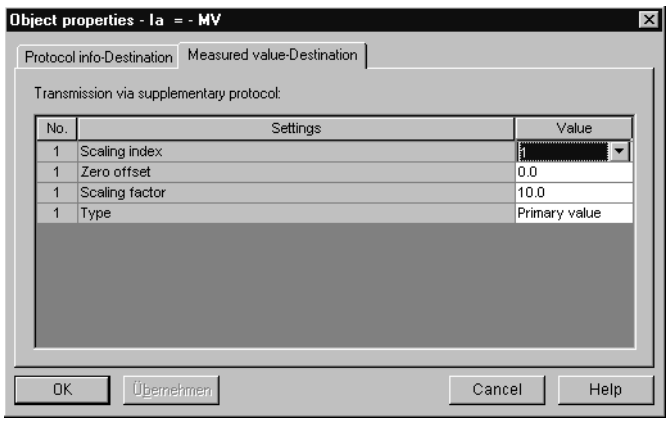


Figure 4-10 Scaling settings of a measured value

The scaling values assigned to the measured values per default are described in the bus mapping documents of the SIPROTEC device types (ref. to page 3).

The scaling - and with that customization to the installation-specific operating values - can be changed by selection of one Scaling index in the **Object properties** dialog window of the measured value.

Scaling index

A predefined scaling possibility (settings of Type, Scaling factor and Zero offset) is summarized using a Scaling index.

Scaling index	Type	Scaling factor	Zero offset
0	Primary value	1.0	0.0
1	Primary value	10.0	0.0
2	Primary value	100.0	0.0
3	Primary value	1000.0	0.0
4	Primary value	10000.0	0.0
5	Secondary value	1000.0	0.0
6	Percentage value	100.0	0.0
7	Secondary value	1.0	0.0
8	Secondary value	10.0	0.0
9	Secondary value	100.0	0.0

Tabelle 4-2 Scaling indices

Note:

The Scaling indices 7 to 9 are not available for every SIPROTEC device.

4.5 Time synchronization

The following Holding registers are relevant for time synchronization of the SIPROTEC devices via Modbus (ref. to page 3, bus mapping documentation of the individual SIPROTEC devices, for the location of these registers):

- Time/Date transfer registers (Time/Date format ref. to chap. 3.7),
- "Set Time and Date" register.

4.5.1 Time/Date transfer from Modbus master

Two methods of data acceptance for time synchronization of the SIPROTEC device via Modbus are possible (ref. to chap. 2.1.1):

Direct writing of time and date

Time and date according to the Time/Date format are transferred completely using a “Preset Multiple Registers” broadcast message (slave address = 0) to the devices.

Time synchronization is executed immediately after reception and evaluation of the Modbus message.

A separate writing of the Time/Date transfer registers (time and date with separate Modbus messages) is not permitted and rejected with Modbus exception 02 (ILLEGAL_DATA_ADDRESS).

The “Set Time and Date” register does not exist in this time synchronization data acceptance mode.
A read or write access to this register is rejected with Modbus exception 02 (ILLEGAL_DATA_ADDRESS).

Use of “Set Time and Date” Register

A separate writing of date and time to the Time/Date transfer registers using Modbus functions “Preset Single Register” or “Preset Multiple Registers” is possible.
This can be done using broadcast messages preferably or with addressed messages to every device.

The time and date which are transferred are the values for the next time synchronization acceptance.

Time synchronization with the current values in the Time/Date transfer registers is executed when the value $FFFF_{hex}$ is written to the "Set Time and Date" register using a broadcast message and Modbus function "Preset Single Register" or "Preset Multiple Registers".

The value 0 is always read from the "Set Time and Date" register.

Note:

- When reading the Time/Date transfer registers the values of time and date written last via Modbus are given back.
- The time interval for the cyclic time synchronization messages from Modbus master (setting time and date directly or writing the “Set Time and Date” register) must be in the range of 50.05 to 60.95 seconds.
  Therefore, “1 minute” is the correct selection in the time master.

4.5.2 Parameterization in DIGSI

Source of time synchronization

In the dialog window **Time Synchronization & Time Format** select ***Fieldbus*** as **Source of time synchronization** for the SIPROTEC device (ref. to Figure 4-11).

Monitoring

The SIPROTEC device continuously monitors the reception of time synchronization messages.

After expiry of the monitoring time **Fault indication after:** (ref. to Figure 4-11) without reception of an time synchronization message via Modbus an error message "Clock SyncError" (Internal object no. = 68) is created.

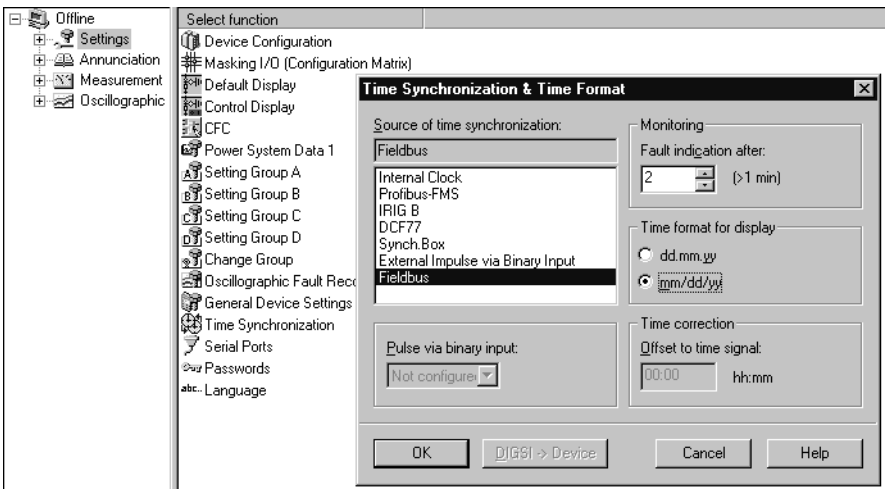


Figure 4-11 Source of time synchronization and Monitoring

Technical data

5

This chapter gives a summary about the technical data of the Modbus slave of the SIPROTEC device including the bus interface.

5.1 Modbus slave of the SIPROTEC devices

Modbus Slave	Slave addresses	1 - 247
	Modbus modes	RTU, ASCII
	Modbus functions	Read Coil Status Read Input Status Read Holding Registers Read Input Registers Force Single Coil Preset Single Register Read Exception Status Diagnostics[1] Subfct. 0 (Return Query Data) Subfct. 2 (Return Diagnostic Reg.) Subfct. 10 (Clear Counters) Subfct. 12 (Return Bus Comm. Error Count) Subfct. 13 (Return Bus Exception Error Count) Subfct. 14 (Return Slave Message Count) Force Multiple Coils Preset Multiple Regs
Data transmission	Baud rates (Bit/s)	300, 600, 1200, 2400, 4800, 9600, 19200 in addition from Modbus version 03.00.04: 38400, 57600
	Parity bit	RTU mode: NONE, EVEN, ODD ASCII mode: EVEN, ODD in addition from Modbus version 04.00.04: RTU and ASCII mode: NONE2 (two Stop bits)

1. Diagnostic subfunctions 10, 12, 13, 14 are available from Modbus firmware version 04.00.04.

5.2 Hardware interface

Two communication modules are available for the connection of Modbus to the SIPROTEC devices:

AME module Universal asynchronous communication module with isolated RS485 interface.

AMO module Universal asynchronous communication module with fibre-optical interface.

5.2.1 Connection via the AME module

Connection	9-pole D-SUB outlet (ref. to Table 5-1)
Protocol	semi-duplex
Max. line length	1000 m / 3300 ft.
Insulation level	500 V_{AC}
Bus termination	On the communication module: integrated, connectable terminating resistors • 221 Ohm between A and B • 392 Ohm between B and VCC1 as well as A and GND1 Input resistance not terminated ≥ 10 kOhm, then bus termination via bus plug with integrated terminating resistors.
Level	Transmitter: • Low: -5 V ≤ U_{A-B} ≤ -1,5 V • High: +5 V ≥ U_{A-B} ≥ +1,5 V Receiver: • Low: U_{A-B} ≤ -0,2 V • High: U_{A-B} ≥ +0,2 V Transmitter and receiver are surge-proof for voltages between A and GND1 as well as between B and GND1 in the range of -7 V...+12 V.
Max. number of modules at the bus	32 For exclusive utilization of AME modules at the bus. This value could be smaller depending on the used Modbus master and further modules at the bus. If more then 32 devices are needed, RS485 repeaters which support bit retiming have to be used.

Bus connection

Pin	Signal	Meaning
1	Shield	Shield / Operational ground
2	-	-
3	A	RS485 connection pin A
4	RTS	Directions control (TTL level)
5	GND1	Data transmission level (ground towards VCC1)
6	VCC1	Supply voltage for terminating resistors (+5V DC, max. 100 mA)
7	-	-
8	B	RS485 connection pin B
9	-	-

Table 5-1 Assignment of the bus connection at the device (D-SUB outlet)

5.2.2 Connection via the AMO module

Anschluss	fibre-optical interface, Rx and Tx, 820 nm, BFOC/2.5 (ST plug)
Protocol	semi-duplex
Max. line length	• 2000 m / 1.25 miles for glass fibre 62.5/125 µm • 3.5 m for plastic fibre
Optical receiver sensitivity	-24 dBm for glass fibre 62.5/125 µm
Optical budget	min. 8 dB for glass fibre 62.5/125 µm
Status for “no signal”	light OFF

Glossary

AME	Universal asynchronous communication module with (electrical) isolated RS485 interface for the SIPROTEC devices from Siemens.
AMO	Universal asynchronous communication module with fibre-optical interface for the SIPROTEC devices from Siemens.
CFC	Continuous Function Chart
CRC	Cyclical Redundancy Check
DC	Double Command
DIGSI	Parameterization system / parameterization software for SIPROTEC devices
DP	Double-point indication
GS	General scan
Input data / Input direction	Data from the Modbus slave to the Modbus master.
LRC	Longitudinal Redundancy Check
LSB	Least Significant Byte
Mapping	Allocation of the SIPROTEC data objects to the positions in the Modbus register map.
MSB	Most Significant Byte
Output data / Output direction	Data from the Modbus master to the Modbus slave.
SC	Single command
SOE	Sequence of Events
SP	Single-point indication

TC	Transformer tap change command
TM	Transformer tap position indication / Transformer tap message

Index

P

Q

R

S

T

V

To

Siemens AG

Dept. PTD PA D DM

D–13623 Berlin

Germany

From

Name:

Company/Dept.:

Address:

Phone no.: Fax no.:

Dear reader,

printing errors can never be entirely eliminated: therefore, should you come across any when reading this manual, kindly enter them in this form together with any comments or suggestions for improvement that you may have.

Corrections/Suggestions

Subject to technical alteration

Siemens Aktiengesellschaft

Order-no.: C53000-L1840-C001-03