Reading Diagnostics Data from a DP slave with SIMATIC S7-1200

SIMATIC S7-1200 and ET 200S

FAQ • November 2012

Siemens - Industry Automation and Drive Technologies - Service& Support - Automation Service, A - Windows Internet Explorer
http://support.automation.siemens.com
Live Search
Siemens - Industry Automation and Drive Techn...
Seite
Extras
SIEMENS
→ siemens.com
→ Industry Automation and Drive Technolog
International | Intranet
Deutsch | Français | Italiano | Español | 中文
Contact | Sitemap
Industry Automation and Drive Technologies Service & Support
Home | Product Support | Applications & Tools | Services | Information | Forum
Login | Register
[Advanced]
Search
Support news
Subscribe to our newsletter now!
XML RSS
world-wide support
Choose a country
my Support
Login
To get registered
My Documentation Manager
Start
Contact
Contacts worldwide
Global Business Coordination
Field Service
Spare parts
Help
Online Help
Guided Tour
Lokales Intranet | Geschützter Modus: Aktiv
100%
Siemens - Industry ...
DE
15:42
SELECT/MENU
MODE
INPUT
EXIT/AUTO

→ SIMOTION supports PROFINET V2.2 according to the specifications of PI (PROFIBUS & PROFINET International) from Version V4.1 SP2 onwards

→ SIMATIC HMI HMI device Mobile Panel 277F IWLAN

Self-help

Search Product Support Documents

Enter your specific product information below to quickly extract the latest related entries from our global database.

Product Name or Part Number: Go

Browse Support Documents

Go to our global database and explore the product specific FAQs, manuals, downloads and approvals/certifications:

→ Product Support

For automation system interaction and connectivity questions, click below for application examples, samples, demonstration systems and more:

→ Applications & Tools

Support by experts worldwide

Technical Forum

View existing conversations and talk to other users about specific product applications and issues.
Please note that you will need to Login to actively participate.

Now start your discussion with other users

Ask the Siemens expert

Send your question directly to a Siemens technical support expert via our online Support Request:

→ Support Request

Our service offer

Repairs, spare parts and consulting

Our Service products are designed to meet your support needs for the entire life cycle of your installation:

→ Spare parts and repairs

→ Technical Consulting

Service & Support

Answers for industry.

SIEMENS

Clicking the link below directly displays the download page of this document.

http://support.automation.siemens.com/WW/view/en/66439168

Caution
The functions and solutions described in this article confine themselves predominantly to the realization of the automation task. Furthermore, please take into account that corresponding protective measures have to be taken in the context of Industrial Security when connecting your equipment to other parts of the plant, the enterprise network or the internet. Further information can be found in Entry ID 50203404.

http://support.automation.siemens.com/WW/view/en/50203404

Question

How do you read the diagnostics data from a PROFIBUS DP slave with S7-1200 using the “DP_NRM_DG” instruction and how do you evaluate the data in the user program?

Answer

The instructions and notes listed in this document provide a detailed answer to this question.

Table of Contents

1 Introduction

Each DP slave has slave diagnostic data that is structured in compliance with EN 50 170 Volume 2, PROFIBUS.

You use the "DPNRM_DG" instruction (read diagnostic data of a DP slave) to read the current diagnostics data of a DP slave in the form defined by EN 50 170 Volume 2, PROFIBUS.

The following table shows the structure of the slave diagnostics data.

Table 1-1

Byte	Description
0	Station status 1
1	Station status 2
2	Station status 3
3	Master station number
4	Manufacturer ID (high byte)
5	Manufacturer ID (low byte)
6…	More slave-specific diagnostics

2 Description of the Sample Program

In the sample program below the diagnostics data of an ET 200S is read out and evaluated with the S7-1200.

The user program consists of the components below.

Table 2-1

Block	Symbolic name	Description
OB1	Main	Cyclic program calls the function block FB2 "DIAG".
OB100	Startup	Startup OB
FB2	DIAG	The function block FB2 "DIAG" calls the "ModuleStates" instruction to read the status information of the DP slave, ET 200S, for example. Depending on the status information of the ET 200S, the "DPNRM_DG" instruction is called to read the diagnostics data of the ET 200S in the format specified in EN 50 170 Volume 2, PROFIBUS.
DB2	iDB_DIAG	Instance data block of the function block DB2 "DIAG"
DB82	Diagnostics Data	The diagnostics data of the ET 200S (slave diagnostics data) is stored in data block DB82.

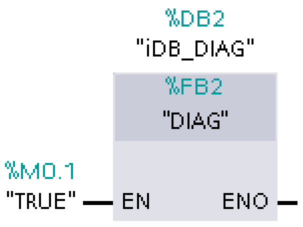
2.1 OB100

The OB100 is a restart OB and is run when the CPU is restarted (warm start). In this OB the variables of the BOOL data type which initiate the request to read the diagnostics data are initialized with the value 1.

2.2 OB1

OB1 is called cyclically. This OB includes the call of the FB2 "DIAG" including instance data block DB2.

Figure 2-1



2.3 FB2

FB2 is called in the OB1 cycle.

2.3.1 Call the "ModuleStates" Instruction

FB2 includes the call of the "ModuleStates" instruction to read the status information of the DP slave, ET 200S, for example.

Inputs

The table below shows the inputs of the "ModuleStates" instruction.

Table 2-2

Input	Data type	Description
LADDR	HW_DEVICE	Identification number of the DP slave (ET 200S)
MODE	UINT	Status type: • 1: The configuration of the module is active or has not yet finished. • 2: Module defective • 3: Module disabled • 4: Module available
STATE	Version	Buffer that receives the error status of the separate modules • Summary bit: Bit 0 = 1 if one of the status bits of the modules is 1. • Status bit: Status of the module with the slot number n as per the selected MODE. If, for example, MODE=2 and Bit 3 = 1, this means that the module in slot 3 is defective.

Outputs

The table below shows the inputs of the "ModuleStates" instruction.

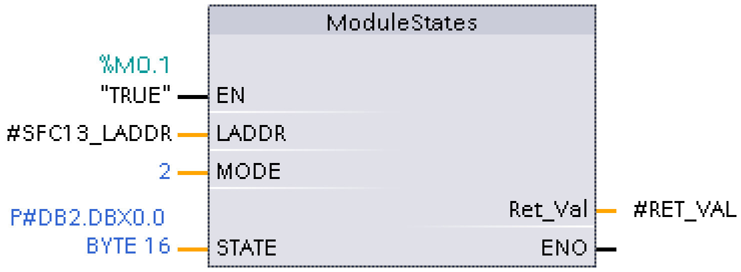
Table 2-3

Output	Data type	Description
RET_VAL	INT	Status of the instruction

Parameterizing the "MODE" input

In this example, the "ModuleStates" instruction is called with the parameter MODE=2. This reads the status information that the DP slave (ET 200S) provides when it is defective.

Figure 2-2



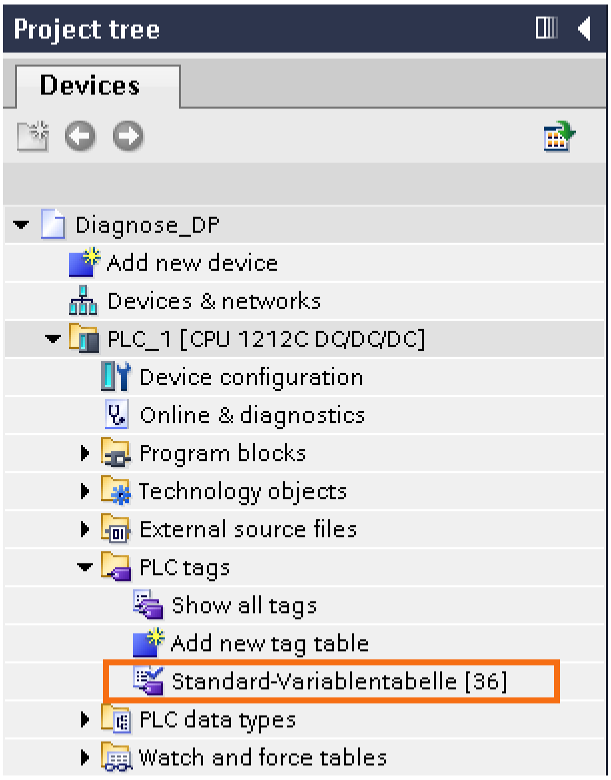
Parameterizing the "LADDR" input

At the LADDR input of the "ModuleStates" instruction you specify the identification number of the DP slave (ET 200S). With the identification number you define the DP slave from which the status information is read.

The identification number of the DP slave (ET 200S) is defined through the configuration and stored in a system constant in the standard tag table.

In the project tree, double-click the "Standard tag table" item to display it in the working area.

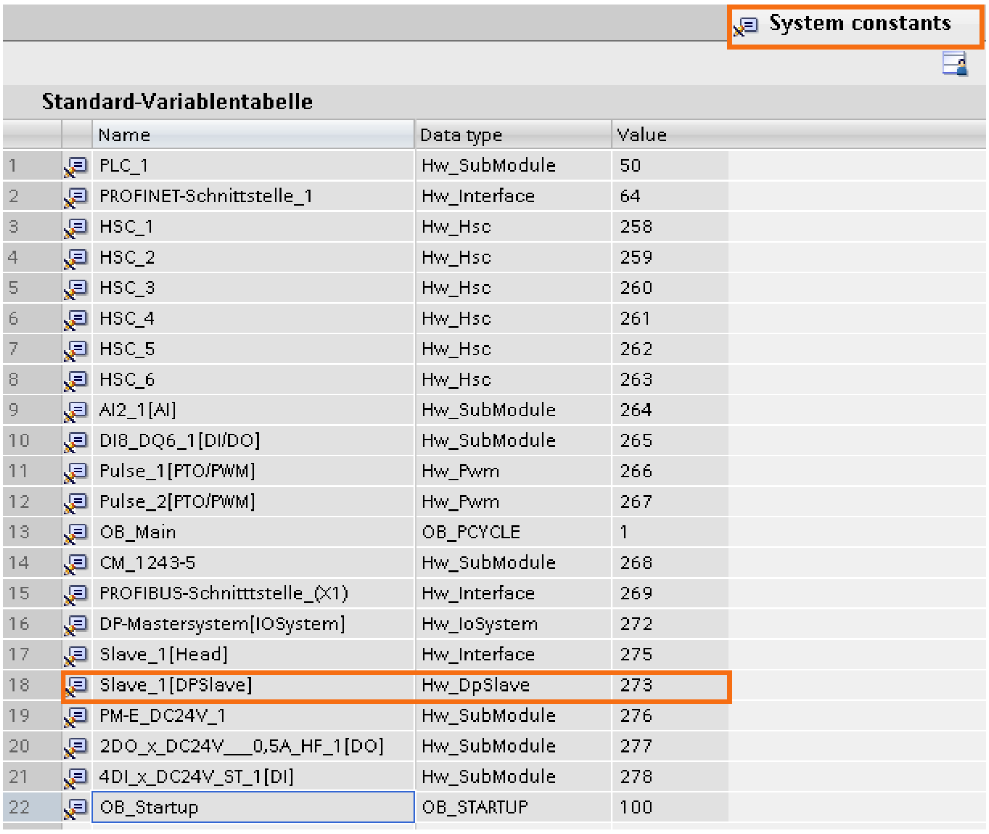
Figure 2-3



Select the "System constants" tab to display all the system constants.

In this example, the identification number of the DP slave (ET 200S) is saved with the value 273 (dec) = w#16#0111 (hex) in system constant Slave_1[DP_Slave].

Figure 2-4



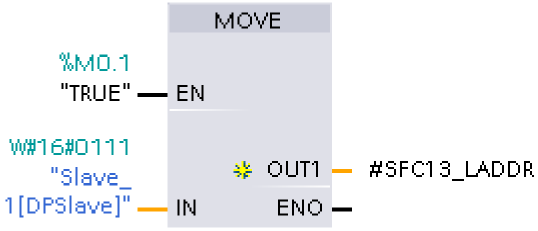
System constants

Standard-Variablentabelle

	Name	Data type	Value
1	PLC_1	Hw_SubModule	50
2	PROFINET-Schnittstelle_1	Hw_Interface	64
3	HSC_1	Hw_Hsc	258
4	HSC_2	Hw_Hsc	259
5	HSC_3	Hw_Hsc	260
6	HSC_4	Hw_Hsc	261
7	HSC_5	Hw_Hsc	262
8	HSC_6	Hw_Hsc	263
9	AI2_1[AI]	Hw_SubModule	264
10	DI8_DQ6_1[DI/DO]	Hw_SubModule	265
11	Pulse_1[PTO/PWM]	Hw_Pwm	266
12	Pulse_2[PTO/PWM]	Hw_Pwm	267
13	OB_Main	OB_PCYCLE	1
14	CM_1243-5	Hw_SubModule	268
15	PROFIBUS-Schnitttstelle_(X1)	Hw_Interface	269
16	DP-Mastersystem[IOSystem]	Hw_IoSystem	272
17	Slave_1[Head]	Hw_Interface	275
18	Slave_1[DPSlave]	Hw_DpSlave	273
19	PM-E_DC24V_1	Hw_SubModule	276
20	2DO_x_DC24V___0,5A_HF_1[DO]	Hw_SubModule	277
21	4DI_x_DC24V_ST_1[DI]	Hw_SubModule	278
22	OB_Startup	OB_STARTUP	100

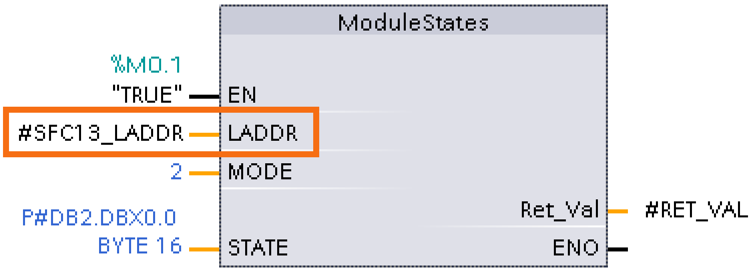
In FB2 "DIAG", the identification number of the DP slave (ET 200S) is saved in the associated instance data block DB2 "iDB_DIAG" in the SFC13_LADDR tag of the WORD data type.

Figure 2-5



You specify the SFC13_LADDR tag at the LADDR inputs of the "ModuleStates" instruction.

Figure 2-6



Parameterizing the "STATE" input

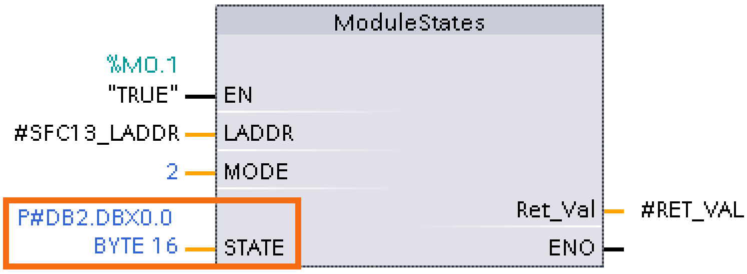
At the STATE input of the "ModuleStates" instruction you specify the memory area in which the status information is stored. You can use any bit sequence (BOOL, BYTE, WORD) as data type or an array of a bit sequence (array of BYTE, for example). The length of the bit sequence depends on the device used. The maximum length is 128 bits.

In this example, the memory area below is defined:

- p#DB2.DBX0.0 BYTE 16

In this way 16 BYTE of status information is stored in the instance data block DB2 of the FB2 starting with address 0.

Figure 2-7



2.3.2 Evaluate the Status Information

When a module of the DP slave (ET 200S) is defective, this triggers the job to read the diagnostics data of the DP slave (ET 200S).

The job to read the diagnostics data is triggered by the following tags of the instance data block DB2 "iDB_DIAG".

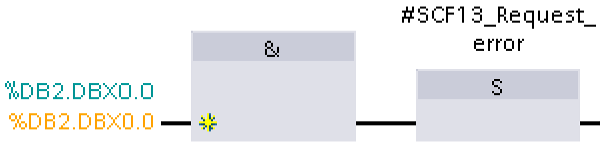
Table 2-4

Name	Data type	Address
SFC13_Request_error	BOOL	DB2.DBX18.0
SFC13_Request_gone	BOOL	DB2.DBX18.3

The Bit 0 of the status information read provides information about the status of the modules of the DP slave. If the Bit 0 of the status information read has the value 1:

- One of the modules at slot 1 to n is defective.
- The SFC13_Request_error tag is set to the value 1.

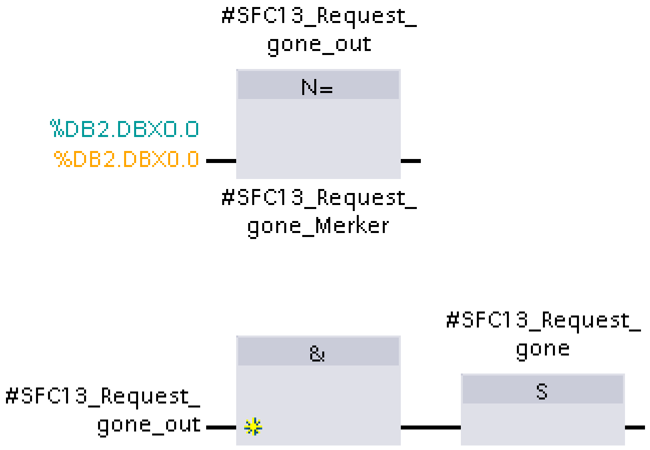
Figure 2-8



When the error of the module at slot 1 to n is cleared:

- The value of the Bit 0 switches from 1 to 0 (negative edge).
- The SFC13_Request_gone is set to the value 1 to read the updated diagnostics data of the DP slave (ET 200S).

Figure 2-9

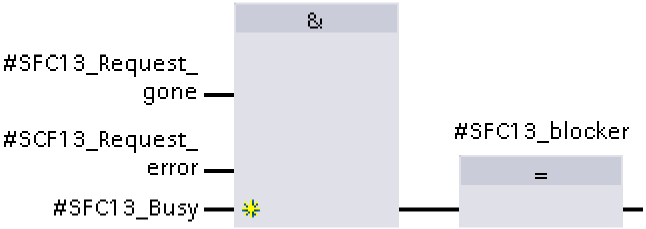


If the error of the module has already been cleared when the job to read the diagnostics data is still running, the BUSY output of the DPNRM_DG as well as the SFC13_Request_error and SFC13_Request_gone tags deliver the value 1.

In the instance data block DB2 "iDB_DIAG", the SFC13_blocker tag of the BOOL data type is assigned the value "1".

In this way the SFC13_Request_error and SFC13_Request_gone tags are not reset in the current block cycle so that another job for reading the diagnostics data is triggered to update it.

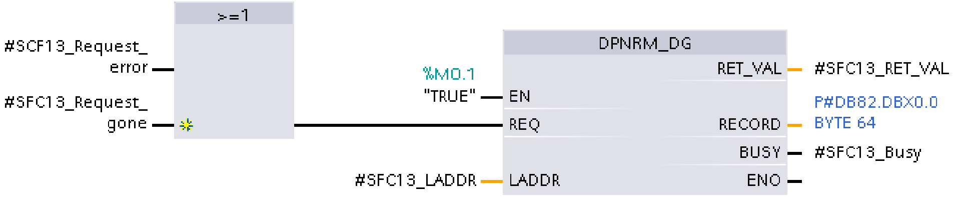
Figure 2-10



2.3.3 Call the "DPNRM_DG" Instruction

FB2 includes the call of the "DPNRM_DG" instruction to read the status information of the DP slave, ET 200S, for example.

Figure 2-11



Inputs

The table below shows the inputs of the "DPNRM_DG" instruction.

Table 2-5

Input	Data type	Description
REQ	BOOL	REQ = 1: Request to read
LADDR	WORD	Configured diagnostics address of the DP slave (identification number of the DP slave)

Outputs

Output	Data type	Description
RET_VAL	INT	If a fault occurs during processing of the instruction, the return value includes the relevant error code. In no fault occurs, in RET_VAL is the length of the data actually transferred.
RECORD	Version	Target area for the diagnostics data read. Only the BYTE data type is permissible. The minimum length of the data record to be read or the target area is 6. The maximum length of the data record to be read is 240.
BUSY	BOOL	BUSY = 1: The read job has not yet finished.

Parameterizing the "REQ" input

The read job is started when the "DPNRM_DG" instruction is called with the input REQ = 1. The REQ has the value 1 when the SFC13_Request_error tag or the SFC13_Request_gone tag has the 1.

Parameterizing the "LADDR" input

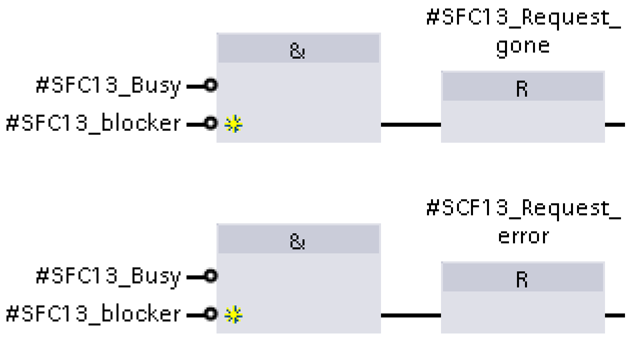
At the LADDR input of the "DPNRM_DG" instruction you specify the SFC13_LADDR tag in which the identification number of the DP slave (ET 200S) is stored.

Parameterizing the "RECORD" output

At the RECORD output of "DPNRM_DG" instruction you specify the target area in which the diagnostics data read is to be stored after data transfer has been completed successfully. In this example the diagnostics data is stored in data block DB82 "DiagnosticsData".

When the reading job is completed, the REQ input once again has the value 0 to enable a new reading procedure to be started.

Figure 2-12



2.3.4 Evaluate the Diagnostics Data

In FB2 "DIAG" the diagnostics data is evaluated which was read out with the "DPNRM_DG" instruction and stored in data block DB82 "DiagnosticsData".

In this example the ET 200S consists of the modules below.

- Module 1: Power module
- Module 2: Digital output module (DO)
- Module 3: Digital input module (DI)

In the user program, evaluation of the diagnostics produces the statuses below for each of the modules mentioned above.

- Module is OK
- Module fault
- Incorrect module
- No module slotted

Structure of the diagnostics data in DB82 "DiagnosticsData"

The table below shows the structure of the diagnostics data that the ET 200S delivers and which is stored in DB82 "DiagnosticsData".

Table 2-6

Byte	Name	Description
0	Stationstatus1	The station statuses 1 to 3 give an overview of the status of the DP slave.
1	Stationstatus2	
2	Stationstatus3	
3	PROFIBUS_Address_Master	PROFIBUS address of the DP master
4	Manufacturer_high_byte	Manufacturer ID
5	Manufacturer_low_byte	
6	specific_diagnostic1	The ID-specific diagnostics indicates whether or not modules of the ET 200S are defective.
7	specific_diagnostic2	
8	specific_diagnostic3	
9	specific_diagnostic4	
10	specific_diagnostic5	
11	specific_diagnostic6	
12	specific_diagnostic7	
13	specific_diagnostic8	
14	specific_diagnostic9	
15	Modulestatus_LEN_CODE	The module status indicates the status of the configure modules and gives details of the ID-specific diagnostics in relation to the configuration.
16	Modulestatus_TYPE	
17	not_relevant1	
18	not_relevant2	
19	Modulestatus_1_4	Status of modules 1 to 4
20	Modulestatus_5_8	Status of modules 5 to 8
21	Modulestatus_9_12	Status of modules 9 to 12
22	Modulestatus_13_16	Status of modules 13 to 16

Byte	Name	Description
23	Modulestatus_17_20	Status of modules 17 to 20
24	Modulestatus_21_24	Status of modules 21 to 24
25	Modulestatus_25_28	Status of modules 25 to 28
26	Modulestatus_29_32	Status of modules 29 to 32
27	Modulestatus_33_36	Status of modules 33 to 36
28	Modulestatus_37_40	Status of modules 37 to 40
29	Modulestatus_41_44	Status of modules 41 to 44
30	Modulestatus_45_48	Status of modules 45 to 48
31	Modulestatus_49_52	Status of modules 49 to 52
32	Modulestatus_53_56	Status of modules 53 to 56
33	Modulestatus_57_60	Status of modules 57 to 60
34	Modulestatus_61_64	Status of modules 61 to 64
35	Channel1_1	The channel-specific diagnostics provides information about channel errors of modules and gives details of the ID-specific diagnostics.
36	Channel1_2	
37	Channel1_3	
38	Channel2_1	
39	Channel2_2	
40	Channel2_3	
41	Channel3_1	
42	Channel3_2	
43	Channel3_3	
44	Channel4_1	
45	Channel4_2	
46	Channel4_3	
47	Channel5_1	
48	Channel5_2	
49	Channel5_3	
50	Channel6_1	
51	Channel6_2	
52	Channel6_3	
53	Channel7_1	
54	Channel7_2	
55	Channel7_3	
56	Channel8_1	
57	Channel8_2	
58	Channel8_3	
59	Channel9_1	
60	Channel9_2	
61	Channel9_3	
62	Reserve1	
63	Reserve2	

In this example, Byte 19 is evaluated to determine the status of modules 1 to 3 of the ET 200S.

The table below shows the structure of Byte 19 and the statuses modules 1 to 3 of the ET 200S can have and report through the diagnostics data.

Table 2-7

Byte 19	Bit number							
	7	6	5	4	3	2	1	0
Module	Module 4		Module 3		Module 2		Module 1	
Value	-	-	0	0	0	0	0	0
Module status	Not applicable		Module OK, valid user data		Module OK, valid user data		Module OK, valid user data	
Value	-	-	0	1	0	1	0	1
Module status	Not applicable		Module error, invalid user data		Module error, invalid user data		Module error, invalid user data	
Value	-	-	1	0	1	0	1	0
Module status	Not applicable		Incorrect module, invalid user data		Incorrect module, invalid user data		Incorrect module, invalid user data	
Value	-	-	1	1	1	1	1	1
Module status	No module (or module failed), invalid user data		No module (or module failed), invalid user data		No module (or module failed), invalid user data		No module (or module failed), invalid user data	

The values of the separate bits of Byte 19 are evaluated in FB2 "DIAG" to determine the status of modules 1 to 3. The status information is stored in the instance data block DB2 "iDB_DIAG" in the following tags.

Table 2-8

Name	Data type	Address
Module1_ok	BOOL	DB2.DBX24.0
Module2_ok	BOOL	DB2.DBX24.1
Module3_ok	BOOL	DB2.DBX24.2
Module1_error	BOOL	DB2.DBX24.3
Module2_error	BOOL	DB2.DBX24.4
Module3_error	BOOL	DB2.DBX24.5
Module1_not_correct	BOOL	DB2.DBX24.6
Module2_not_correct	BOOL	DB2.DBX24.7
Module3_not_correct	BOOL	DB2.DBX25.0
Module1_not_plug_in	BOOL	DB2.DBX25.1
Module2_not_plug_in	BOOL	DB2.DBX25.2
Module3_not_plug_in	BOOL	DB2.DBX25.3