SSL-Lists with PROFINET/PROFIBUS

PROFIBUS, PROFINET

Compendium • December 2011

Applikationen & Tools

Answers for industry.

SIEMENS

Industry Automation and Drive Technologies Service & Support Portal

This article is taken from the Service Portal of Siemens AG, Industry Automation and Drive Technologies. The following link takes you directly to the download page of this document:

http://support.automation.siemens.com/WW/view/en/24000238

Caution

The functions and solutions described in this article confine themselves to the realization of the automation task predominantly. Please take into account furthermore that corresponding protective measures have to be taken up 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 under the Content-ID 50203404.

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

If you have any questions concerning this document please e-mail us to the following address:

mailto:online-support.industry@siemens.com

You can also actively use our Technical Forum from the Service & Support Portal regarding this subject. Add your questions, suggestions and problems and discuss them together in our strong forum community:

http://www.siemens.com/forum-applications

S

SIMATIC
System Status Lists

Warranty and Liability

Note

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These application examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice.
If there are any deviations between the recommendations provided in these application examples and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.

Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.

Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of Siemens Industry Sector.

Preface

Aim of this document

This document is aimed at the SIMATIC programmers who want to program a profound diagnosis of their system during runtime with the in every SIMATIC S7 CPU implemented system status lists. The System Status Lists (SSL) are listed in the form of a compendium.

Validation

This document is valid for all products of the PROFIBUS-DP and PROFINET IO environment.

Table of Contents

1 System Status Lists in PROFINET IO and PROFIBUS DP

Content of this section

This section covers:

- The system status lists designed for PROFINET IO
- The system status lists designed for PROFIBUS DP
- The system status lists designed for PROFINET IO and PROFIBUS DP

Introduction

The CPU of SIMATIC modules can provide specific information. The CPU saves this information to the "system status list" (SSL).

The SSL describes the current status of the automation system. It provides an overview of:

- the configuration
- the current parameter settings
- the current states
- processes in the CPU and the modules assigned to it

The data in the system status list is read-only. The SSL is a virtual list which is only created on request.

You can use the SSL to obtain the following information about the PROFINET IO system:

- System data
- Module status information in the CPU
- Diagnostics data for a module
- Diagnostics buffer

Compatibility of the new SSLs

New system status lists were implemented for PROFINET IO, as PROFINET supports larger volumes of project data.

You should also use these new SSLs with PROFIBUS.

You can use an existing PROFIBUS SSL as usual if this list is also supported in PROFINET. If you try to use an SSL which is not supported in PROFINET, the program returns an error code in RET_VAL (8083: Incorrect or illegal index).

Comparison of the SSLs of PROFINET IO and PROFIBUS DP

Table 1-1 Comparison of the SSLs of PROFINET IO and PROFIBUS DP

SZL-ID	PROFINET IO	PROFIBUS DP	Validity
W#16#0591	Yes (parameter adr1 changed)	Yes	Module status information for the interfaces of a module/submodule
W#16#0A91	Yes (parameter adr1 changed)	Yes	Status information of all subsystems and master systems (S7-300 without CPU 318-2 DP only)
W#16#0C91	Yes (parameter adr1/adr2 and target/actual type ID changed)	Yes	Module status information of a module/submodule in a central rack or interconnected with an integrated DP or PN interface module using the logical module address.
W#16#4C91	No	Yes	Not for S7-300 Module status information of a module interconnected with an external DP or PN interface module using the start address
W#16#0D91	Yes (parameter adr1 changed)	Yes	Module status information of all modules in the specified rack/station
W#16#0696	Yes	No	Module status information of all submodules of a module using the logical address of the module; not possible for submodule 0 (= module)
W#16#0C96	Yes	Yes	Module status information of a submodule using the logical address of this submodule
W#16#xy92	No (replaced with: SSL ID W#16#0x94)	Yes	Rack/station status information Also replace this SSL in PROFIBUS DP with the SSL with ID W#16#xy94.
W#16#0x94	Yes	Yes	Target state of the stations or central racks
W#16#x294	Yes	Yes	Actual state of the stations or central racks
W#16#0x694	Yes	Yes	All stations of an IO subsystem or all central racks in error state
W#16#0x794	Yes	No	Error/maintenance state of the stations or central racks

Additional information about SSLs

For detailed information about the individual SSLs, refer to the *System Software for S7-300/400 System and Standard Functions manual* (see Point \4\ in Chapter 4) and the STEP 7 V5.4 SP1 online help.

2 Records with PROFINET IO

Content of this section

This section covers:

- The most important differences between PROFINET IO and PROFIBUS DP in terms of diagnostics
- How the diagnostics mechanism works in PROFINET IO
- Structure of a diagnostics and configuration record in PROFINET IO.

Additional information

For further information about diagnostics, refer to the STEP 7 Online Help.

2.1 Introduction

2.1.1 Overview of diagnostics and configuration records

Consistent diagnostics concept

PROFINET IO supports you with a consistent diagnostics concept.

The next section describes the basic features of the diagnostics concept.

Diagnostics mechanism

The IO device outputs a diagnostics interrupt to the IO controller when it detects faults such as wire break. This interrupt calls a corresponding OB in the user program (diagnostics interrupt OB82), in order to generate a defined (programmed) response to the fault.

The IO controller automatically sets the parameters and configures a replacement device or module. Cyclic exchange of user data is restored in the next step.

Diagnostics records in PROFINET IO

There are two different types of diagnostics record:

1. Channel diagnostics records

   Channel diagnostics records are generated if a channel is in an error state and/ or has triggered an interrupt.

   A diagnostics record of length 0 is returned if there is no fault.

2. Vendor-specific diagnostics records

   The structure and size of vendor-specific diagnostics records depend on the vendor's settings.

   For information about vendor-specific diagnostics records, refer to the appropriate device manual.

Profile and structure of the diagnostics and configuration records

A PROFINET IO device consists of one or more "logical devices". These devices in turn contain one or more APIs (Application Process Identifiers) and at least API 0. The PROFINET IO profile (PROFIdrive, for example) is coded using the API.

Each PROFINET IO device supports at least one **A**pplication **P**rocess **I**dentifier (API).

The diagnostics records (W#16#800A, for example) can differ in terms of their structure. This difference is characterized by the BlockVersion. BlockVersion 0101 of record W#16#X00A, for example, was upgraded with the API number in order to enable diagnostics for IO devices with several APIs.

By contrast to PROFIBUS DP, the API is used in PROFINET IO as a profile ID and is a parameter which identifies the profile. Examples for various application scenarios:

Table 2-1 Application profiles

Field of application	Profile	API
Drive technology	PROFIdrive	W#16#3A00 - W#16#3AFF
Safety technology	PROFIsafe	W#16#3E00 - W#16#3EFF
Conveyor systems	Intelligent Pumps	W#16#5D00 - W#16#5DFF

Requirements

The diagnostics information is only generated for configured modules / submodules / channels.

List of diagnostics and configuration records in PROFINET IO

The table below lists important diagnostics records in PROFINET IO.

The specified record sizes apply to at least one faulty channel.

Table 2-2 Diagnostics records in PROFINET IO

Record number	Content and meaning	Size in bytes
W#16#800A	The record returns - channel diagnostics data and / or - extended channel diagnostics data for a submodule slot. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#800B	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a submodule slot. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#800C	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a submodule slot. Note: This record is only available if maintenance is requested or demanded, and when a fault is detected; refer to section 2.5.7. The record may also contain status information for an IE/PB Link.	0 - 4176
W#16#8010	The record returns - channel diagnostics data and / or - extended channel diagnostics data for a submodule slot. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#8011	The record returns - channel diagnostics data and / or - extended channel diagnostics data for a submodule slot. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176

Record number	Content and meaning	Size in bytes
W#16#8012	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a submodule slot. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#8013	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a submodule slot. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#C00A	The record returns - channel diagnostics data and / or - extended channel diagnostics data for a module slot. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#C00B	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a module slot. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#C00C	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a module slot. Note: This record is only available if maintenance is requested or demanded, and when a fault is detected; refer to section 2.5.7. The record may also contain status information for an IE/PB link.	0 - 4176
W#16#C010	The record returns - channel diagnostics data and / or - extended channel diagnostics data for a module slot Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176

Record number	Content and meaning	Size in bytes
W#16#C011	The record returns - channel diagnostics data and / or - extended channel diagnostics data for a module slot. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#C012	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a module slot. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#C013	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a module slot. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#E00A	The record returns - channel diagnostics data and / or - extended channel diagnostics data for an AR. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#E00B	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an AR. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#E00C	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an AR. Note: This record is only available if maintenance is requested or demanded, and when a fault is detected; refer to section 2.5.7. The record may also contain status information for an IE/PB link.	0 - 4176

Record number	Content and meaning	Size in bytes
W#16#E010	The record returns - channel diagnostics data and / or - extended channel diagnostics data for an AR. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#E011	The record returns - channel diagnostics data and / or - extended channel diagnostics data for an AR. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#E012	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an AR. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#E013	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an AR. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#F00A	The record returns - channel diagnostics data and / or - extended channel diagnostics data for an API. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176
W#16#F00B	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an API. Note: This record is only available if a fault is detected; see section 2.5.7.	0 - 4176

Record number	Content and meaning	Size in bytes
W#16#F00C	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an API. Note: This record is only available if maintenance is requested or demanded, and when a fault is detected; refer to section 2.5.7. The record may also contain status information for an IE/PB link.	0 - 4176
W#16#F010	The record returns - channel diagnostics data and / or - extended channel diagnostics data for an API. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#F011	The record returns - channel diagnostics data and / or - extended channel diagnostics data for an API. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#F012	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an API. Note: This record is only available if maintenance is requested; see section 2.5.7.	0 - 4176
W#16#F013	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for an API. Note: This record is only available if maintenance is demanded; see section 2.5.7.	0 - 4176
W#16#F80C	The record returns - channel diagnostics data and / or - extended channel diagnostics data and / or - vendor-specific diagnostics data for a device. Note: This record is only available if maintenance is requested or demanded, and when a fault is detected; refer to section 2.5.7. The record may also contain status information for an IE/PB link.	0 - 4176

The list below shows the important configuration records in PROFINET IO.

Table 2-3 Configuration data records in PROFINET IO

Record number	Content and meaning	Size in bytes
W#16#8000	Target configuration at subslot level	22 - 4176
W#16#C000	Target configuration at subslot level	22 - 4176
W#16#E000	Target configuration at AR level	22 - 4176
W#16#8001	Actual configuration at subslot level	0 - 4176
W#16#C001	Actual configuration at subslot level	0 - 4176
W#16#E001	Actual configuration at AR level	0 - 4176
W#16#E002	Deviation from the target configuration of the IO device	0 - 4176
W#16#F000	Actual configuration at API level	0 - 4176

Structure of other records

The structure of all the records is defined in the *PROFINET IO - Application Layer Service Definition - Application Layer Protocol Specification* standard. Members of the PROFIBUS User Organisation can download this standard from the homepage (see chapter 4).

A list of vendor IDs is included in the management information of OB82 for PROFINET IO and is also available at the homepage of the PROFIBUS User Organisation (see chapter 4).

Additional information

For further information about diagnostics, refer to the PROFINET system description system manual.

2.1.2 Overview of the additional records in PROFINET IO

Overview of the records relevant to PROFINET IO

Table 2-4 Records for reading I/O handling in PROFINET IO

Record number	Content and meaning	Size in bytes
W#16#801E	This record returns the substitute values for a submodule.	0 - 4176
W#16#8028	This record returns the current input data of the submodule.	0 - 4176
W#16#8029	This record returns the current output data of the submodule	0 - 4176

Table 2-5 Records which return the status of the PROFINET interfaces

Record number	Content and meaning	Size in bytes
W#16#802A	This record returns the current port settings.	0 - 4176
W#16#802B	This record returns the configured port settings.	0 - 4176
W#16#802F	This record returns the configured port settings.	0 - 4176
W#16#8060	This record returns the current settings of the optical port.	0 - 4176
W#16#8061	This record returns the configured settings of the optical port.	0 - 4176
W#16#8062	This record returns the configured settings of the optical port.	0 - 4176
W#16#8070	This record returns the configured settings of the PROFINET interface.	0 - 4176
W#16#F831	This record returns the group record for the configured settings of the PROFINET interface and its ports (IRT parameter settings only).	0 - 4176
W#16#F841	This record returns the group record for the current settings of the PROFINET interface and its ports.	0 - 4176
W#16#F842	This record returns the group record for the configured settings of the PROFINET interface and its ports.	0 - 4176

Table 2-6 Records for reading/writing I&M data in PROFINET IO

Record number	Content and meaning	Size in bytes
W#16#AFF0	This record returns I&M 0 data.	0 - 4176
W#16#AFF1	This record returns I&M 1 data.	0 - 4176
W#16#AFF2	This record returns I&M 2 data.	0 - 4176
W#16#AFF3	This record returns I&M 3 data.	0 – 4176
W#16#F840	This record returns a list of submodules which send different I&M 0 data.	0 – 4176

Table 2-7 Records for reading/writing protocol parameters in PROFINET IO

Record number	Content and meaning	Size in bytes
W#16#F821	This record returns all APIs supported by a PROFINET IO device.	0 - 4176
W#16#F830	This record returns a list of internal error events, for example, the cause of the cancellation of a communication relationship.	0 - 4176

Additional information

For detailed information about the records, refer to the V2.1 of the PROFINET specification "Application Layer services for decentralized periphery and distributed automation" and "Application Layer protocol for decentralized periphery and distributed automation".

2.2 Device model with PROFINET IO

2.2.1 Device model of an IO device

Introduction

The PROFINET IO device model describes the structure of modular and compact field devices. It builds upon the basic features of PROFIBUS DP.

The definition of submodules and APIs has been added to the device model in order to increase the flexibility of an IO device.

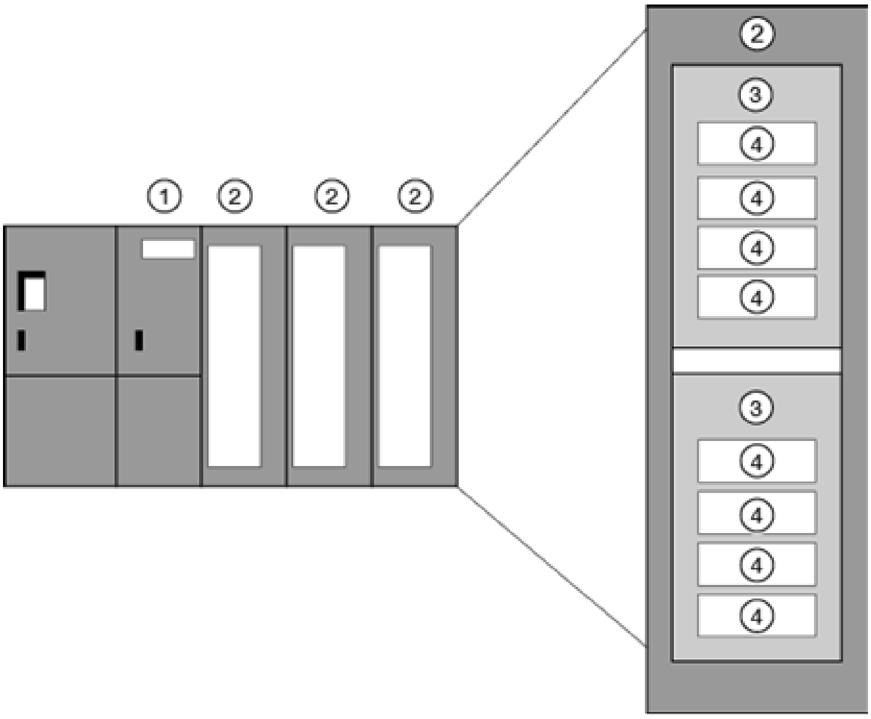
Modules / submodules / channels

A PROFINET IO device has a modular structure similar to a PROFIBUS DP slave.

The modules are inserted into slots and the submodules are inserted into subslots. The modules / submodules have channels which are used to read and output process signals.

The diagram below shows the setup.

Figure 2-1 Structure of a PROFINET device



Digit	Description
①	Slot with interface circuit
②	Slot with module
③	Subslot with submodule
④	Channel

It is possible to divide a slot into several subslots into which the submodules are inserted.

2.2.2 Diagnostics levels with PROFINET IO

Concept

The IO device sends all error messages that occur to the IO controller. The scope and volume of diagnostics information varies according to the level of diagnostics data evaluation.

Diagnostics levels

You can evaluate diagnostics data at different levels.

The number and type of channels is selected at the diagnostics levels.

Figure 2-2 Diagnostic levels in PROFINET IO

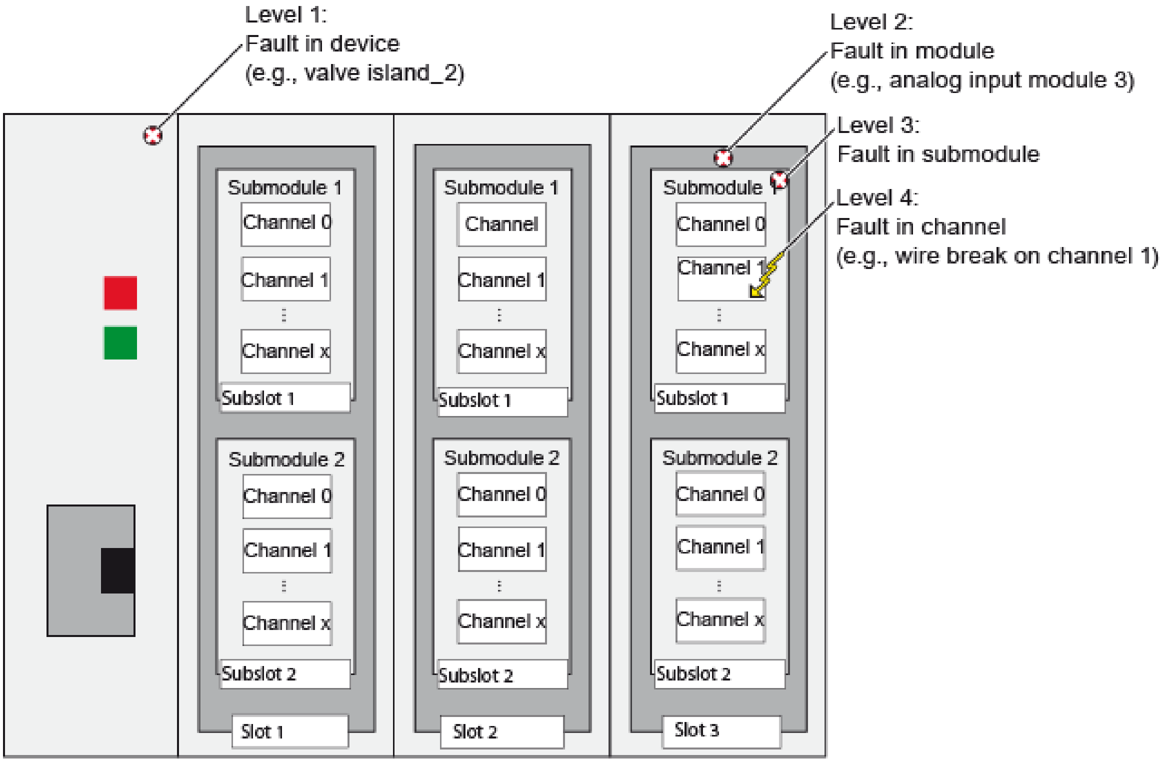


Table 2-8 Diagnostics levels

Level	Fault location
1	Fault in device, valve block 2
2	Fault in module, analog module 3
3	Fault in submodule
4	Channel fault, wire break at channel 1

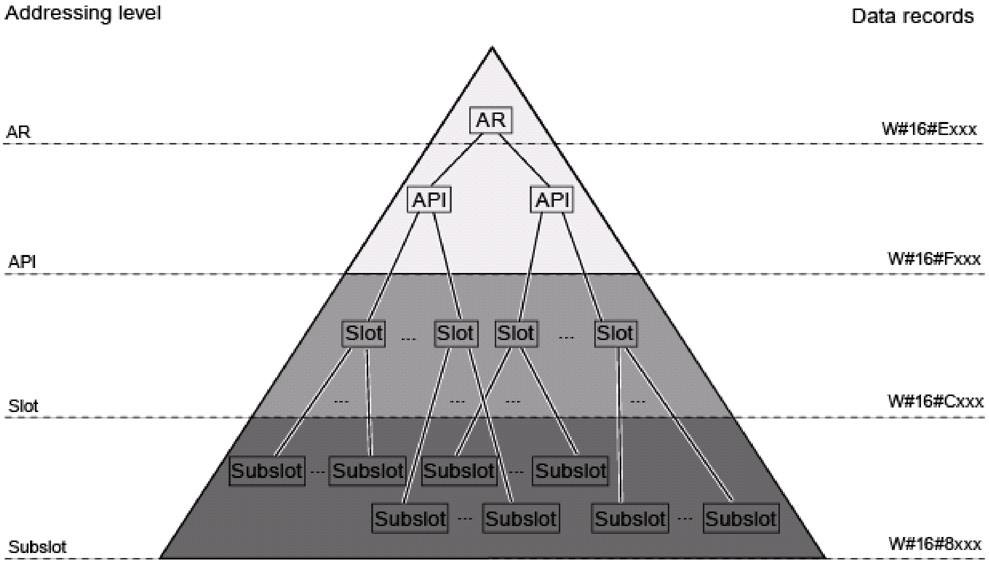
Addressing level and records

Diagnostics and configuration data is evaluated at the following addressing levels:

- AR (Application Relation),
- API (Application Process Identifier),
- Slot
- Subslot

A group of diagnostics and configuration records are available at each addressing level. The individual groups of records are distinguished by the first letter of the record number.

Figure 2-3 Diagnostics levels



The information for each IO device (addressing level AR), module (addressing level slot) or submodule (addressing level subslot) is always transferred in separate diagnostics or configuration records. The record returns diagnostics or configuration data for one or more subslots, slots and APIs, depending on the addressing level.

2.3 Structure of Diagnostics Data Records

2.3.1 Function and selection of a diagnostics data record

Introduction

The basic **structure** is the same for each of the following diagnostics records:

- W#16#800A, W#16#800B, W#16#800C, W#16#8010, W#16#8011, W#16#8012, W#16#8013,
- W#16#C00A, W#16#C00B, W#16#C00C, W#16#C010, W#16#C011, W#16#C012, W#16#C013,
- W#16#E00A, W#16#E00B, W#16#E00C, W#16#E010, W#16#E011, W#16#E012, W#16#E013,
- W#16#F00A, W#16#F00B, W#16#F00C, W#16#F010, W#16#F011, W#16#F012, W#16#F013.

The **content** and **size** of a diagnostics record may vary depending on the type of diagnostics (see block User Structure Identifier).

Record identifier

You can select a suitable record for specific diagnostics in an application program with reference to the record name. The structure is described below.

This relates to the **first digit** and **last two digits** of the record number:

- **First digit**:

The first digit of the name of a diagnostics record (for example, W#16#800A) refers to the **addressing level** (AR, API, slot, subslot). You can request diagnostics information at one of these addressing levels.

- **Last two digits**:

The last two digits of the name of a diagnostics record (W#16#C012) in combination with the **U**ser **S**tructure **I**dentifier (**USI**) identify the type of diagnostics record, such as:

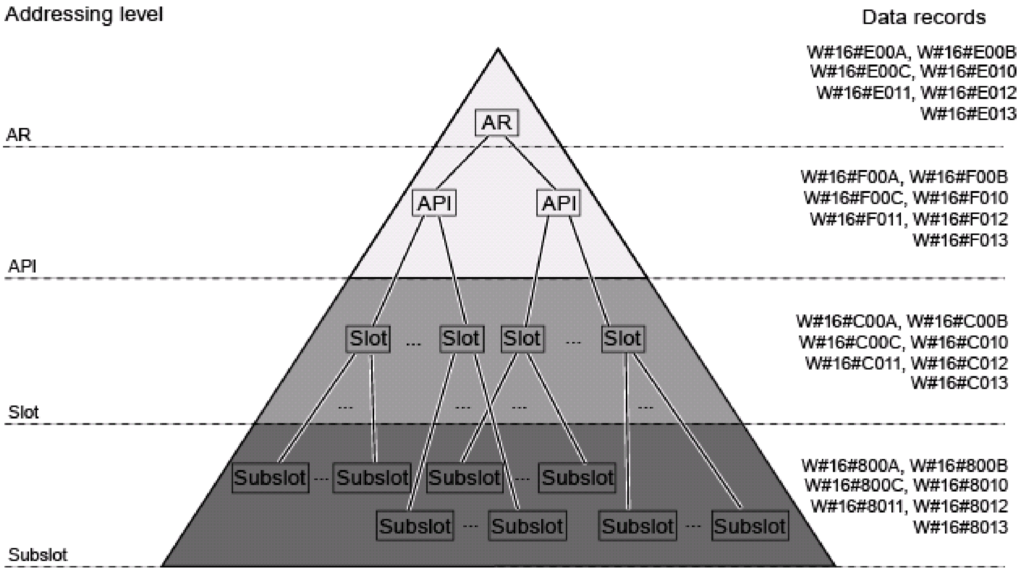
- Channel diagnostics
- Extended channel diagnostics
- Vendor-specific diagnostics
- MaintenanceRequest
- MaintenanceDemanded

Addressing level

There are suitable groups of diagnostics records available for each addressing level - AR, API, slot, subslot - at which diagnostics data is requested.

The first letter of the record number identifies the group (W#16#**E**0XX, W#16#**F**0XX, W#16#**C**0XX or W#16#**8**0XX).

Figure 2-4 Addressing levels for diagnostics records



User Structure Identifier (USI)

The USI identifies the type of diagnostics data:

- Channel diagnostics
- Extended channel diagnostics
- Vendor-specific channel diagnostics

The additional **U**ser **S**tructure **I**dentifier (USI) feature can be used to distinguish between different diagnostics records.

Figure 2-5 Diagnostics record and User Structure Identifier (USI)

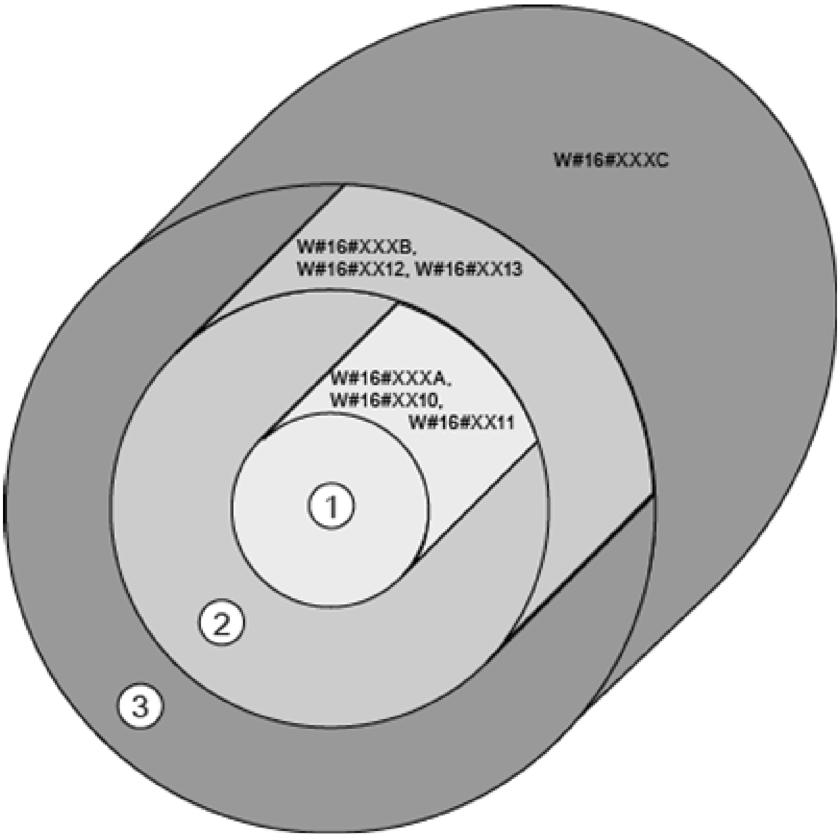


Table 2-9 Diagnostic record and User Structure Identifier (USI)

Digit	Meaning
①	- USI = W#16#8000 -> channel diagnostics - USI = W#16#8002 -> extended channel diagnostics Note: The record may contain channel diagnostics data and extended channel diagnostics data with MaintenanceRequest, MaintenanceDemanded and error messages.
②	- USI = W#16#0000-W#16#7FFF -> vendor-specific diagnostics - USI = W#16#8000 -> channel diagnostics - USI = W#16#8002 -> extended channel diagnostics Note: The record may contain channel diagnostics data, extended channel diagnostics data and vendor-specific diagnostics data with MaintenanceRequest, MaintenanceDemanded and error messages.
③=①+ ②	- USI = W#16#0000-W#16#7FFF -> vendor-specific diagnostics - USI = W#16#8000 to W#16#80FF -> range for standardized USI values, for example, W#16#8000 (channel diagnostics) or W#16#8002 (extended channel diagnostics) - USI = W#16#9000 to W#16#9FFF -> profile-specific Note: The records may contain MaintenanceRequest, MaintenanceDemanded and error messages.

Example

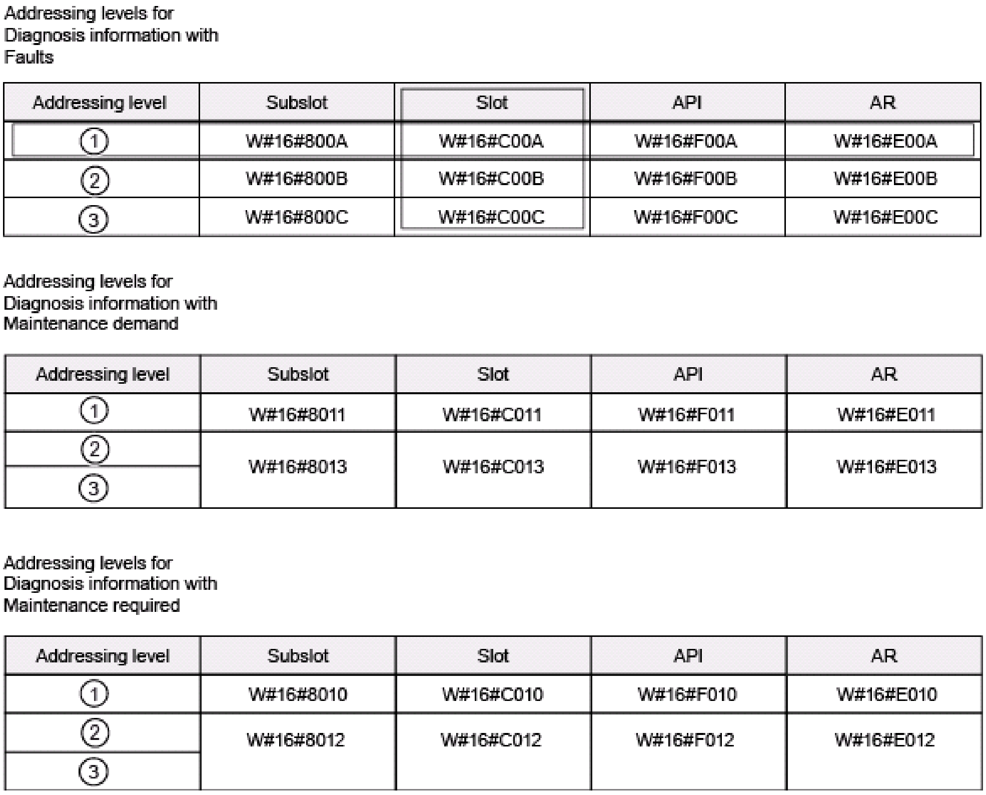
The example below shows how to select a suitable record.

You only evaluate the **channel diagnostics data** of a **slot** on the ET 200S. Use Table 2-4 to select the suitable record as follows:

1. **Channel diagnostics data** is contained in all diagnostics records with the **USI=W#16#8000 / W#16#0x8002** (see diagram above). This concerns all diagnostics records with record number W#16#X0**0A**
   (framed row in Table seen at Figure 2-6 Addressing levels and records).
2. The diagnostics function is called at **slot** level, which suggests all diagnostics records with the numbers W#16#**C**0XX (framed column in Table seen at Figure 2-6 Addressing levels and records) for selection.

You select diagnostics record **W#16#C00A** which contains all available channel diagnostics data of the slot (module).

Figure 2-6 Addressing levels and records



Addressing levels for Diagnosis information with Faults

Addressing level	Subslot	Slot	API	AR
①	W#16#800A	W#16#C00A	W#16#F00A	W#16#E00A
②	W#16#800B	W#16#C00B	W#16#F00B	W#16#E00B
③	W#16#800C	W#16#C00C	W#16#F00C	W#16#E00C

Addressing levels for Diagnosis information with Maintenance demand

Addressing level	Subslot	Slot	API	AR
①	W#16#8011	W#16#C011	W#16#F011	W#16#E011
②	W#16#8013	W#16#C013	W#16#F013	W#16#E013
③				

Addressing levels for Diagnosis information with Maintenance required

Addressing level	Subslot	Slot	API	AR
①	W#16#8010	W#16#C010	W#16#F010	W#16#E010
②	W#16#8012	W#16#C012	W#16#F012	W#16#E012
③				

See above for the key to digits ① to ③.

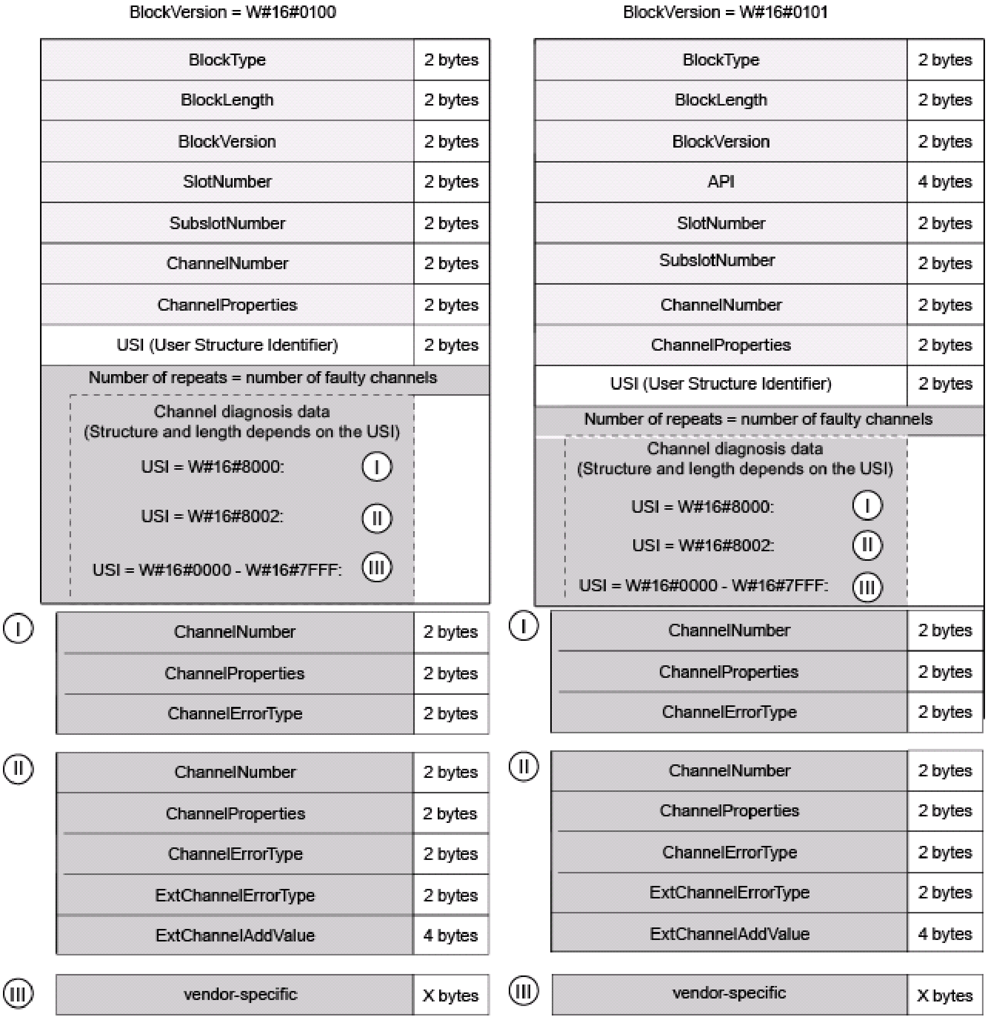
2.3.2 Structure of diagnostics data records

Block diagram of the structure

The diagram shows the structure of the diagnostics records with their DBs:

- W#16#800A, W#16#800B, W#16#800C, W#16#8010, W#16#8011, W#16#8012, W#16#8013
- W#16#C00A, W#16#C00B, W#16#E00C, W#16#C010, W#16#C011, W#16#C012, W#16#C013
- W#16#E00A, W#16#E00B, W#16#E00C, W#16#E010, W#16#E011, W#16#E012, W#16#E013
- W#16#F00A, W#16#F00B und W#16#F00C, W#16#F010, W#16#F011, W#16#F012, W#16#F013

Figure 2-7 Diagnostics records



BlockVersion = W#16#0100

BlockType	2 bytes
BlockLength	2 bytes
BlockVersion	2 bytes
SlotNumber	2 bytes
SubslotNumber	2 bytes
ChannelNumber	2 bytes
ChannelProperties	2 bytes
USI (User Structure Identifier)	2 bytes
Number of repeats = number of faulty channels	

Channel diagnosis data
(Structure and length depends on the USI)
USI = W#16#8000: (I)
USI = W#16#8002: (II)
USI = W#16#0000 - W#16#7FFF: (III)	

(I)

ChannelNumber	2 bytes
ChannelProperties	2 bytes
ChannelErrorType	2 bytes

(II)

ChannelNumber	2 bytes
ChannelProperties	2 bytes
ChannelErrorType	2 bytes
ExtChannelErrorType	2 bytes
ExtChannelAddValue	4 bytes

(III)

vendor-specific	X bytes

BlockVersion = W#16#0101

BlockType	2 bytes
BlockLength	2 bytes
BlockVersion	2 bytes
API	4 bytes
SlotNumber	2 bytes
SubslotNumber	2 bytes
ChannelNumber	2 bytes
ChannelProperties	2 bytes
USI (User Structure Identifier)	2 bytes
Number of repeats = number of faulty channels	

Channel diagnosis data
(Structure and length depends on the USI)
USI = W#16#8000: (I)
USI = W#16#8002: (II)
USI = W#16#0000 - W#16#7FFF: (III)	

(I)

ChannelNumber	2 bytes
ChannelProperties	2 bytes
ChannelErrorType	2 bytes

(II)

ChannelNumber	2 bytes
ChannelProperties	2 bytes
ChannelErrorType	2 bytes
ExtChannelErrorType	2 bytes
ExtChannelAddValue	4 bytes

(III)

vendor-specific	X bytes

The diagnostics record with BlockVersion W#16#0100 is located on the left of the diagram; the diagnostics record with BlockVersion W#16#0101 is located on the right. It also contains the Application Process Identifier (API). The API data field shows the coding of all available profiles (PROFIdrive, for example).

The structure of the ChannelDiagnosisData block is based on the User Structure Identifier. All three possible structures are identified by the selection boxes

(I) – (III)

API

Application **P**rocess **I**dentifier. Term used in PROFINET IO Standard IEC 61158; its value specifies the application which is used to process IO data.

The IEC assigns the APIs specific profiles which are defined in context by the PROFINET User Organisation. The standard API is 0.

Number of faulty channels

General rule: A diagnostics record (ChannelDiagnosisData) is generated for each subslot with faulty channels. A record of length 0 will be sent if no fault is detected.

Several instances of the ChannelDiagnosisData DB are present if several channel faults are detected.

You can determine the number of faulty channels by reading the data value in the BlockLength data field (number of successive bytes). Comparison with the values in the table header identifies the relevant column, while the USI with the BlockVersion gives the row containing the number of faulty channels.

If the BlockLength = 28 for a diagnostics record with USI= W#16#8000 and BlockVersion= W#16#0101, for example, then the table (column 6, row 3) shows that **2 channels** are faulty.

Table 2-10 ChannelDiagnosis and number of channels

USI	Block Version	BlockLength							
		18 Byte	22 Byte	24 Byte	28 Byte	30 Byte	34 Byte	36 Byte	40 Byte
W#16 #8000	W#16#0 001	1 channel	-	2 channels	-	3 channels	-	4 channels	-
	W#16#0 101	-	1 channel	-	**2 channels example** (see also section 3.3)	-	3 channels	-	4 channels
W#16 #8002	W#16#0 101	-	-	1 channel	-	-	-	2 channels	-
	W#16#0 101	-	-	-	1 channel	-	-	-	2 channels

Data blocks, detailed information

For detailed information about the various DBs of the diagnostics records, refer to section 0.

2.3.3 Evaluating diagnostics data

Task definition

You want to evaluate diagnostics and status data of an IO device in the application program.

The example below demonstrates the evaluation of diagnostics data record W#16#E00C.

Note the following information:

- Chapter 3.3: The description of the procedure is based on the example of diagnostics data record W#16#E00C.
- Chapter 2.3.2: Structure of diagnostics data records

General procedure

1. Read diagnostics data record W#16#E00C by calling SFB52.
2. Evaluate the LEN parameter of SFB52 -> Result: LEN = 58.
3. Read the values of the following parameters of the diagnostics data record:
   - BlockLength in bytes 2 and 3 -> Result: BlockLength = W#16#001C; converted = 28 bytes
   - BlockVersion in bytes 4 and 5 -> Result: BlockVersion = W#16#0101
   - USI for BlockVersion W#16#0101 in bytes 18 and 19 -> Result: USI = W#16#8000 -> Returns 6 bytes of channel diagnostics data per faulty channel.

Result:

Reading the value of BlockLength and knowledge of the structure of diagnostics record W#16#E00C for BlockVersion W#16#0101 with USI W#16#8000 gives the following result:

The record with a total length of 32 bytes contains two channel diagnostics records.

	BlockLength = W#16#001C = 16 + 6 + 6 = 28		
BlockType + BlockLength	**BlockVersion … USI**	**Diagnostics data for channel 1**	**Diagnostics data for channel 0**
Total length of this record = 4 Bytes + 16 Bytes + 6 Bytes + 6 Bytes = 32 Bytes			

Additional diagnostics data is available for evaluation because LEN > 32 bytes.

4. Again, read the values of the following parameters of the second diagnostics record:
   - BlockLength in byte 34 and 35 -> Result: BlockLength = W#16#0016; converted = 22 bytes
   - BlockVersion in byte 36 and 37 -> Result: BlockVersion = W#16#0101
   - USI in bytes 50 and 51 for BlockVersion W#16#0101 -> Result: USI = W#16#8000 -> Returns 6 bytes of channel diagnostics data per faulty channel.

Result:

Reading the value of BlockLength and knowledge of the record structure of diagnostics record W#16#E00C for BlockVersion 0101 with USI W#16#8000 returns the following result:

The record with a total length of 26 bytes contains two channel diagnostics records.

	BlockLength = 16 Bytes + 6 Bytes = 22 Bytes	
BlockType + **BlockLength**	**BlockVersion …** **USI**	**Diagnosedaten für Kanal 0**
Total length of this record = 4 Bytes + 16 Bytes + 6 Bytes = 26 Bytes		

Overall result

The first record has a length of 32 bytes, the second has a length of 26 bytes. The length total of both records is 58 bytes. This value is consistent with parameter LEN = 58 bytes and indicates that all the data has been evaluated and that no other data exists.

2.4 Structure of the configuration data records

2.4.1 Structure of the configuration data records W#16#8000, W#16#8001, W#16#C000, W#16#C001, W#16#E000, W#16#E001, W#16#E002, W#16#F000

Introduction

The basic **structure** of configuration records W#16#8000, W#16#8001, W#16#C000, W#16#C001, W#16#E000, W#16#E001, and W#16#F000 is the same.

The **content** and **size** of a configuration record differs according to the type of configuration.

Record identifier

You can select a suitable record for specific configuration information in a user program with reference to the record names. Their structure is described below.

This relates to the **first digit** and **last digit** of the name of a configuration record:

- **First digit - addressing level:**

The first digit of the name of a configuration record (for example, W#16#8001) identifies **the addressing level** (AR, API, slot, subslot). Configuration data can be requested at this addressing level.

- **Last digit - target / actual configuration:**

If this is **0** as in W#16#800**0**:

then it is a configuration record that you can use to request the **target configuration**.

If it is **1** as in W#16#800**1**:

then it is a configuration record that you can use to request the **actual configuration.**

Note

Configuration record W#16#F000

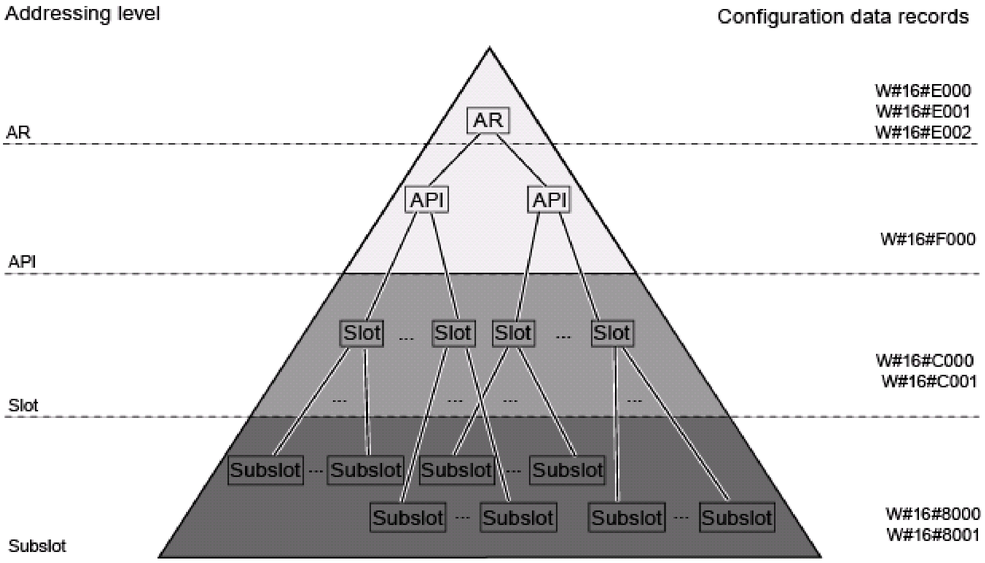
Only configuration record W#16#F000 is used to request the actual configuration. It is therefore an exception in the identification scheme described above.

Addressing level

The model of a PROFINET IO device describes its modular structure, which is based on the general principles of PROFIBUS DP (refer to PROFINET System Manual, pages 2 to 6).

You can use suitable groups of configuration records to request configuration data at a specific addressing level, that is, AR, API, slot or subslot. The first digit of the configuration record identifies the group.

Figure 2-8 Addressing levels and associated configuration records



Example

The following example explains how to select a suitable record. You want to read the **actual configuration** from a **slot** at an ET 200S.

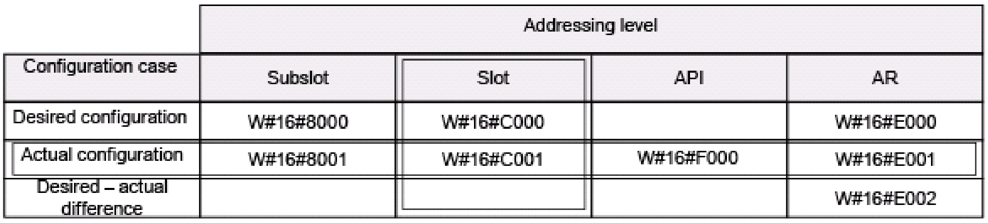
To select the right configuration record from the table of Figure 2-9 Configuration records for the target and actual configuration):

1. Read the **actual configuration** using the configuration record with a "1" as the last digit of the record number, that is, W#16#8001, W#16#C001, W#16#E001 and W#16#F000 (third row in the table of Figure 2-9 Configuration records for the target and actual configuration).

2. The configuration is addressed at the **slot** level (third column of the table of Figure 2-9 Configuration records for the target and actual configuration).

3. This returns record **W#16#C001** which you can use to read the actual configuration of any slot.

Figure 2-9 Configuration records for the target and actual configuration



	Addressing level			
Configuration case	Subslot	Slot	API	AR
Desired configuration	W#16#8000	W#16#C000		W#16#E000
Actual configuration	W#16#8001	W#16#C001	W#16#F000	W#16#E001
Desired – actual difference				W#16#E002

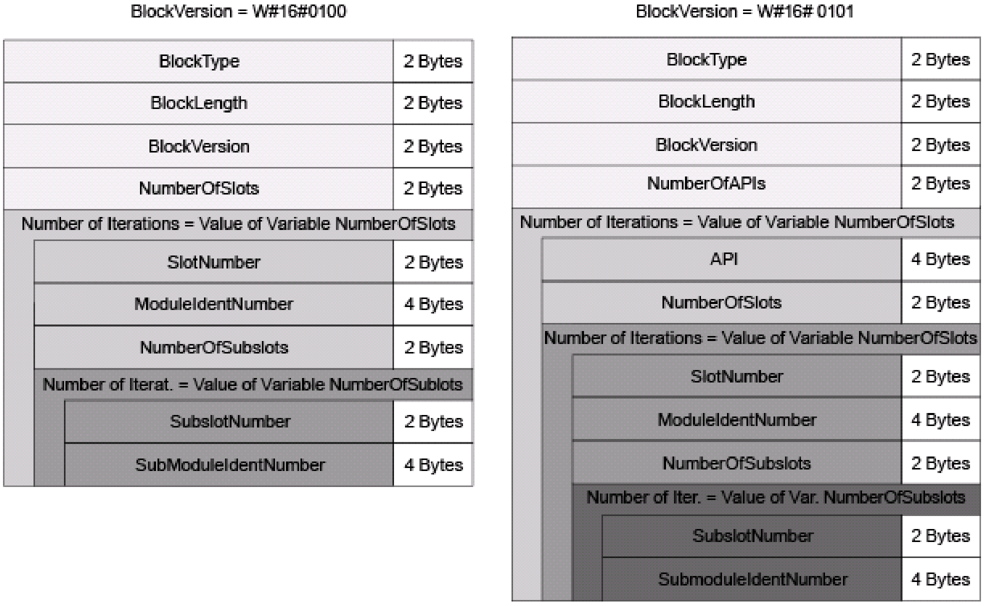
Block diagram of the structure

The diagram below shows the structure of the configuration records:

- W#16#8000, W#16#8001
- W#16#C000, W#16#C001
- W#16#E000, W#16#E001
- W#16#F000

Including their data blocks.

Figure 2-10 PROFINET IO Record W#16#8000, W#16#8001, W#16#C000, W#16#C001, W#16#E000, W#16#E001, W#16#F000



BlockVersion = W#16#0100

BlockType	2 Bytes
BlockLength	2 Bytes
BlockVersion	2 Bytes
NumberOfSlots	2 Bytes
Number of Iterations = Value of Variable NumberOfSlots	
SlotNumber	2 Bytes
ModuleIdentNumber	4 Bytes
NumberOfSubslots	2 Bytes
Number of Iterat. = Value of Variable NumberOfSublots	
SubslotNumber	2 Bytes
SubModuleIdentNumber	4 Bytes

BlockVersion = W#16# 0101

BlockType	2 Bytes
BlockLength	2 Bytes
BlockVersion	2 Bytes
NumberOfAPIs	2 Bytes
Number of Iterations = Value of Variable NumberOfSlots	
API	4 Bytes
NumberOfSlots	2 Bytes
Number of Iterations = Value of Variable NumberOfSlots	
SlotNumber	2 Bytes
ModuleIdentNumber	4 Bytes
NumberOfSubslots	2 Bytes
Number of Iter. = Value of Var. NumberOfSubslots	
SubslotNumber	2 Bytes
SubmoduleIdentNumber	4 Bytes

The configuration record with BlockVersion W#16#0100 is located on the left of the diagram, and the configuration record with BlockVersion W#16#0101 is located on the right.

Configuration records with BlockVersion W#16#0101 support multiple APIs and also include the Application Process Identifier (API). The API data field shows the coding of all available profiles (PROFIdrive, for example).

Data blocks, detailed information

For detailed information about the various DBs of the diagnostics records, refer to section 0.

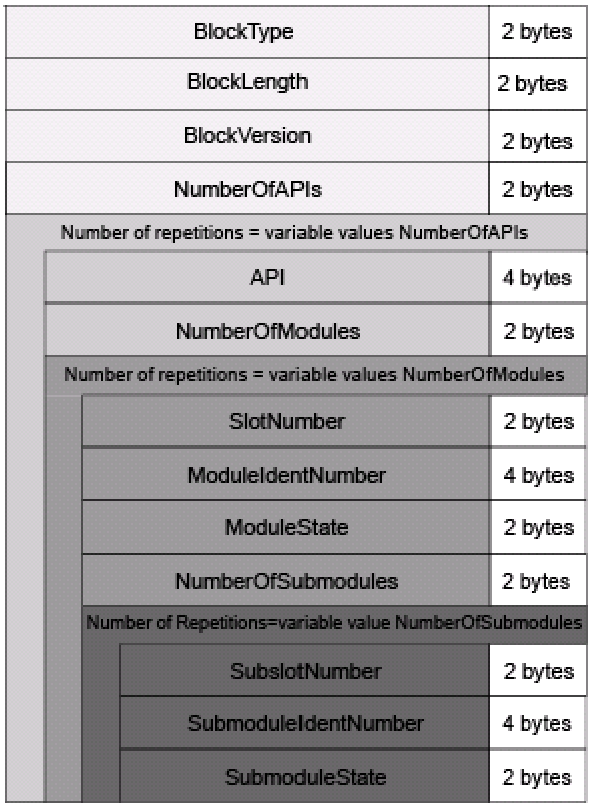
2.4.2 Structure of the configuration data record W#16#E002

Schematic Structure

The following graphic shows the structure of the configuration data record W#16#E002 and the composition from the data blocks shown.

The data blocks are described in the chapter 0.

Figure 2-11 PROFINET IO Record W#16#E002



BlockType	2 bytes
BlockLength	2 bytes
BlockVersion	2 bytes
NumberOfAPIs	2 bytes
Number of repetitions = variable values NumberOfAPIs	
API	4 bytes
NumberOfModules	2 bytes
Number of repetitions = variable values NumberOfModules	
SlotNumber	2 bytes
ModuleIdentNumber	4 bytes
ModuleState	2 bytes
NumberOfSubmodules	2 bytes
Number of Repetitions=variable value NumberOfSubmodules	
SubslotNumber	2 bytes
SubmoduleIdentNumber	4 bytes
SubmoduleState	2 bytes

2.5 Blocks of the diagnostics and configuration records

2.5.1 API

Table 2-11 Adress space of API

API (hexadecimal value)	Meaning
W#16#0000	Default value
W#16#00000001 - W#16#FFFFFFFF	Address space for defined profiles

2.5.2 BlockLength

The BlockLength data field contains the coding of the **number of successive bytes** of the diagnostics or configuration record. This code returns the length of the diagnostics or configuration record, without the number of bytes for the BlockType and BlockLength data fields, which each have a length of 2 bytes.

2.5.3 BlockType

Table 2-12 BlockType coding

BlockType	Meaning
W#16#0001	Interrupt transfer channel 1
W#16#0002	Interrupt transfer channel 2
W#16#0010	Diagnostics record
W#16#0012	Configuration record for target configuration
W#16#0013	Configuration record for actual configuration
W#16#8104	Configuration record Target/actual comparison

2.5.4 BlockVersion

The BlockVersion data block in turn consists of the BlockVersionHigh and BlockVersionLow DBs, which are each one byte long.

Table 2-13 Coding of BlockVersion

BlockVersion	Value (hexadezimal)	Meaning
BlockVersionHigh	B#16#01	Identifies the first value of the version number, W#16#0**1**xx
BlockVersionLow	B#16#00 oder B#16#01	Version number W#16#010**0** or W#16#010**1**

2.5.5 ChannelErrorType

Table 2-14 Coding of ChannelErrorType

Wert (hexadecimal)	Meaning	Error Message
W#16#0000	Reserved	Unknown error
W#16#0001	Short-circuit	Short-circuit
W#16#0002	Undervoltage	Undervoltage
W#16#0003	Overvoltage	Overvoltage
W#16#0004	Overload	Overload
W#16#0005	Overtemperature	Overtemperature
W#16#0006	Wire break	Wire break
W#16#0007	Violation of high limit	Violation of high limit
W#16#0008	Violation of low limit	Violation of low limit
W#16#0009	Error	Error
W#16#000A - W#16#000F	Reserved	Unknown error
W#16#0010	Vendor-specific Incorrect parameter assignment	Incorrect parameter assignment
W#16#0011	Vendor-specific Power supply failure	Power supply failure
W#16#0012	Vendor-specific Fuse blown/tripped	Fuse blown/tripped
W#16#0013	Vendor-specific	Vendor-specific
W#16#0014	Vendor-specific Ground fault	Ground fault
W#16#0015	Vendor-specific Reference point not found	Reference point not found
W#16#0016	Vendor-specific Sampling error	Sampling error
W#16#0017	Vendor-specific Violation of threshold limits	Violation of threshold limits
W#16#0018	Vendor-specific Output switched off	Output switched off
W#16#0019	Vendor-specific Safety-related fault	Safety-related fault
W#16#001A	Vendor-specific External fault	External fault
W#16#001B - W#16#001F	Vendor-specific	Vendor-specific
W#16#0020 - W#16#00FF	Reserved for standard profiles for all devices	Standard profiles for all devices (for example, PROFIsafe)
W#16#0100 - W#16#7FFF	Vendor-specific	Vendor-specific
W#16#8000	Data transfer not possible	Data transfer not possible
W#16#8001	Wrong neighborhood	Wrong neighborhood
W#16#8002	Redundancy loss	Redundancy loss
W#16#8003	Loss of synchronization (on the bus side)	Loss of synchronization (on the bus side)

Wert (hexadecimal)	Meaning	Error Message
W#16#8004	Loss of isochronous operation (at the device)	Loss of isochronous operation (at the device)
W#16#8005	Direct communication error	Direct communication error
W#16#8006	Reserved	Reserved
W#16#8007	Fiber-optics Error	Optical transfer not possible
W#16#8008	Error Network component	Problems with network function
W#16#8009	Timebase error	No timer or problems with timebase accuracy
W#16#800A - W#16#8FFF	Reserved	Unknown error
W#16#9000 - W#16#9FFF	Reserved for technological profiles (for example, PROFIdrive)	Profile-specific
W#16#A000 - W#16#FFFF	Reserved	Unknown error

2.5.6 ChannelNumber

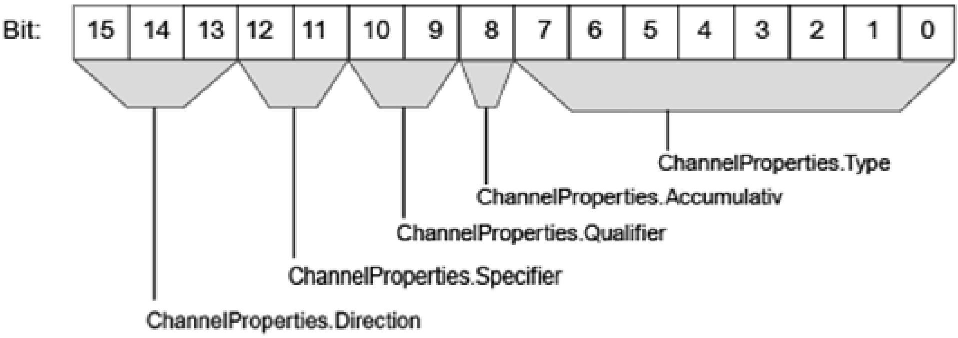
Table 2-15 ChannelNumber coding

Value (hexadecimal)	Meaning
W#16#0000 - W#16#7FFF	Vendor-specific
W#16#8000	Submodule
W#16#8001 - W#16#FFFF	Reserved

2.5.7 ChannelProperties

Struktur von ChannelProperties

Figure 2-12 Structure of ChannelProperties


Bit:
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
ChannelProperties.Type
ChannelProperties.Accumulativ
ChannelProperties.Qualifier
ChannelProperties.Specifier
ChannelProperties.Direction

ChannelProperties.Type (Bit 0 – 7)

Table 2-16 Coding of ChannelProperties.Type

Value (hexadecimal)	Meaning
B#16#00	If ChannelNumber is W#16#8000 (submodule).
B#16#01	1 bit
B#16#02	2 bits
B#16#03	4 bits
B#16#04	8 bits
B#16#05	16 bits
B#16#06	32 bits
B#16#07	64 bits
B#16#08 - B#16#FF	Reserved

ChannelProperties.Accumulative (Bit 8)

Table 2-17 Coding of ChannelProperties.Accumulative

Value (hexadecimal)	Name	Meaning
0	-	No channel error group message (concerns only one channel)
1	Accumulative	Channel error group message (concerns several channels)

Combination of ChannelPropertes.Qualifier (Bit 9/10) and ChannelProperties.Specifier (Bit 11/12)

Table 2-18 Combination of the vlaues of MaintenanceRequired / MaintenanceDemanded and Specifier

Maintenance Required (Bit 9)	MaintenanceDemanded (Bit 10)	Specifier (Bit 12/11)	Meaning	Possible at
0	0	00	All sublevel* diagnostics - MaintenanceRequired, MaintenanceDemanded and Qualified Diagnosis - are no longer active	Evaluation of diagnostics interrupts by calling SFB54 in OB82
		01	Diagnostics active	Evaluation of diagnostics interrupts using SFB54 in OB82 or by calling SFB52 to read the record
		10	Diagnostics no longer active	Evaluation of diagnostics interrupts by calling SFB54 in OB82

Maintenance Required (Bit 9)	MaintenanceDemanded (Bit 10)	Specifier (Bit 12/11)	Meaning	Possible at
		11	Status message - only available in combination with vendor-specific error	Evaluation of diagnostics interrupts by calling SFB54 in OB82
0	1	00	Reserved	—
		01	MaintenanceDemanded active	Evaluation of diagnostics interrupts using SFB54 in OB82 or by calling SFB52 to read the record
		10	MaintenanceDemanded no longer active	Evaluation of diagnostics interrupts by calling SFB54 in OB82
		11	MaintenanceDemanded no longer active - all others remain active	
1	0	00	Reserved	—
		01	MaintenanceRequired active	Evaluation of diagnostics interrupts using SFB54 in OB82 or by calling SFB52 to read the record
		10	MaintenanceRequired no longer active	Evaluation of diagnostics interrupts by calling SFB54 in OB82
		11	MaintenanceRequired no longer active - all others remain active	
1	1	00	Reserved	—
		01	Graded diagnostics active	Evaluation of diagnostics interrupts using SFB54 in OB82 or by calling SFB52 to read the record
		10	Graded diagnostics no longer active	Evaluation of diagnostics interrupts by calling SFB54 in OB82
		11	Graded diagnostics no longer active - all others remain active	

* Sublevel in this context refers to the clearance of all characteristic data of the ExtChannelErrorType and ChannelErrorType blocks from outgoing events.

ChannelProperties.Specifier (Bit 11 - 12)

Table 2-19 ChannelProperties.Specifier coding

Value (hexadecimal)	Meaning	Possible at
00	Reserved	--
01	Diagnostics pending	Evaluation of diagnostics interrupts using SFB54 in OB82 or by calling SFB52 to read the record
10	Outgoing event and no other events	Evaluation of diagnostics interrupts by calling SFB54 in OB82
11	Outgoing event but others remain	Evaluation of diagnostics interrupts by calling SFB54 in OB82

ChannelProperties.Direction (Bit 13 - 15)

Table 2-20 ChannelProperties.Direction coding

Value	Meaning
000	Vendor-specific
001	Input
002	Output
003	Input/output
004 - 007	Reserved

2.5.8 ExtChannelAddValue

This field is of the Unsigned32 data type.

The content of this data field is 0 if information for extended channel diagnostics is not available.

2.5.9 ExtChannelErrorType

Table 2-21 ExtChannelErrorType coding

Value (hexadecimal)	Meaning
W#16#0000 – W#16#FFFF	Coding depends on the ChannelErrorType See the PROFINET IO Application Layer Service Definition & Application Layer Protocol Specification or IEC 61158.

Table 2-22 Coding of ExtChannelErrorType for ChannelErrorType W#16#0000-W#16#7FFF

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Group message
W#16#8001 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-23 Coding of ExtChannelErrorType for ChannelErrorType "No data transfer possible"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Port status error - cable not connected, for example
W#16#8001	Error due to incorrect interface setting – full duplex and half duplex
W#16#8002	Error due to runtime delay – the configured cable length does not match the real cable length
W#16#8003 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-24 Coding of ExtChannelErrorType for ChannelErrorType "wrong neighbourhood"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Wrong neighbor - device
W#16#8001	Wrong neighbor - port
W#16#8002	Neighbor does not support RealTime Class 3 or is not configured
W#16#8003	Error due to incorrect interface setting – full duplex and half duplex
W#16#8004	incorrect or missing configuration of media redundancy
W#16#8005	No neighbor available
W#16#8006	Neighbor does not support bumpless media redundancy

Value (hexadecimal)	Meaning
W#16#8007 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-25 coding of ExtChannelErrorType for ChannelErrorType "Redundancy loss"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Media Redundancy Manager reports error
W#16#8001	Ring open – media redundancy no longer available
W#16#8002	Ring open – bumpless media redundancy no longer available
W#16#8003	Several media redundancy managers in the ring
W#16#8004 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-26 Coding of ExtChannelErrorType for ChannelErrorType "Loss of isochrone mode" and "Timebase error"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	No synchronization received
W#16#8001	RealTime Class 3 – incorrect synchronization configuration
W#16#8002	RealTime Class 3 – incorrect configuration
W#16#8003	Jitter out of limits
W#16#8004 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-27 coding of ExtChannelErrorType for ChannelErrorType "Isochrone state error"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Transfer timeout at outputs
W#16#8001	Transfer timeout at outputs
W#16#8002 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-28 Coding of ExtChannelErrorType for ChannelErrorType "Multicast CR error"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Data recipient in communication - wrong or missing sender
W#16#8001	Data recipient in communication – unknown sender
W#16#8002 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-29 Coding of ExtChannelErrorType for ChannelErrorType "optical transfer not possible“

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Undershoot of specified receiving level
W#16#8001 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

Table 2-30 Coding of ExtChannelErrorType for ChannelErrorType "Network function error"

Value (hexadecimal)	Meaning
W#16#0000	Reserved
W#16#0001 – W#16#7FFF	Vendor-specific
W#16#8000	Network overload - discarding message frames
W#16#8001 – W#16#8FFF	Reserved
W#16#9000 – W#16#9FFF	Reserved for profiles
W#16#A000 – W#16#FFFF	Reserved

2.5.10 ModuleIdentNumber

Table 2-31 ModuleIdentNumber coding

Value (hexadecimal)	Meaning
DW#16#00000000	Reserved
DW#16#00000001 - DW#16#FFFFFFFF	Vendor-specific

2.5.11 ModuleState

Table 2-32 ModuleState coding

Value (hexadecimal)	Meaning	Description
W#16#0000	Module not found	Module not inserted
W#16#0001	Incorrect module	Incorrect ModuleIdentNumber
W#16#0002	Correct module	Module is OK, but at least one submodule is disabled, incorrect or missing - or a submodule is running diagnostics
W#16#0003	Substitute	Module is not the one requested - but is compatible. The IO device is capable of adapting itself to the module
W#16#0004-W#16#FFFF	Reserved	

2.5.12 SlotNumber

Table 2-33 SlotNumber coding

Value (hexadecimal)	Meaning
W#16#0000 - W#16#7FFF	The first slot number is zero. The last slot number is W#16#7FFF.
W#16#8000 - W#16#FFFF	Reserved

2.5.13 SubmoduleIdentNumber

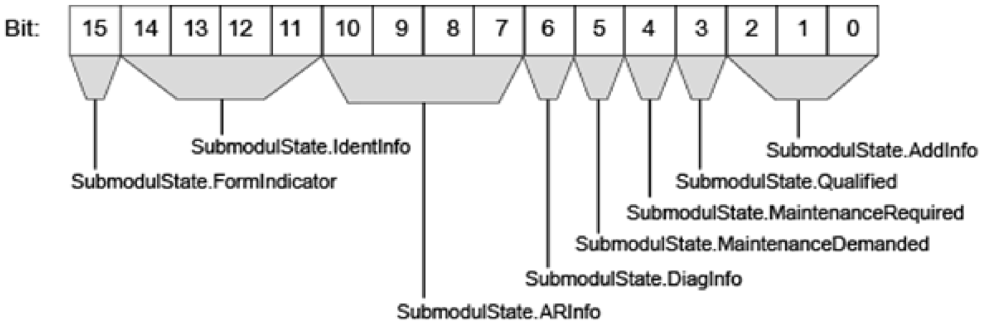
Table 2-34 SubmoduleIdentnumber coding

Value (hexadecimal)	Meaning
DW#16#00000000 - DW#16#FFFFFFFF	Vendor-specific

2.5.14 SubmoduleState

Struktur von SubmoduleState

Figure 2-13 Strucuture of SubmoduleState



SubmoduleState.AddInfo (Bit 0 - 2)

Table 2-35 SubmoduleState.AddInfo coding

Value	Meaning	Description
000	No meaning	
001	Acceptance not allowed	Submodule is not suitable for acceptance from IO supervisor AR
002	Reserved	

SubmoduleState.MaintenanceRequired (Bit 4)

Table 2-36 SubmoduleState.MaintenanceRequired coding

Value (hexadecimal)	Meaning	Description
0	No MaintenanceRequired active	MaintenanceRequired not available for this submodule.
1	MaintenanceRequired available	MaintenanceRequired is available for this submodule.

SubmoduleState. MaintenanceDemanded (Bit 5)

Table 2-37 SubmoduleState.MaintenanceDemanded coding

Value (hexadecimal)	Meaning	Description
0	No MaintenanceDemanded active	MaintenanceDemanded not available for this submodule.
1	MaintenanceDemanded available	MaintenanceDemanded available for this submodule.

SubmoduleState.DiagInfo (Bit 6)

Table 2-38 SubmoduleState.DiagInfo coding

Value (hexadecimal)	Meaning	Description
0	No diagnostics data available	No diagnostics data available or saved for this submodule.
1	Diagnostics data available	Diagnostics data available for this submodule: The data can be read using the corresponding records.

SubmoduleState.ARInfo (Bit 7 - 10)

Table 2-39 SubmoduleState.ARInfo coding

Value	Meaning	Description
0000	User	Submodule is available to the user
0001	ApplicationReadyPending	Submodule is not available to the user, for example, on account of parameter errors
0002	Locked	Submodule is not available to the user. Example: simultaneous request of several overriding functions
0003	Locked by IO controller	The controller is not the owner of the submodule (submodule is not available to the user)
0004	Locked by IO supervisor	The controller is not the owner of the submodule (submodule is not available to the user)
0005 - 000F	Reserved	Reserved

SubmoduleState.IdentInfo (Bit 11 - 14)

Table 2-40 SubmoduleState.IdentInfo coding

Value	Meaning
0000	OK
0001	Substitute
0002	Incorrect
0003	No submodule
0004 - 000F	Reserved

SubmoduleState.FormIndicator (Bit 15)

Table 2-41 SubmoduleState. FormatIndicator coding

Value (hexadecimal)	Meaning	Description
1	SubmoduleState consists of SubmoduleState.IdentInfo, .ARInfo and .AddInfo	Supported by the IO controller, IO device and IO supervisor.
0	Reserved	Reserved

2.5.15 SubslotNumber

Table 2-42 SubslotNumber coding

Value (hexadecimal)	Meaning
W#16#0000	Determined by the module; does not address the submodule.
W#16#0001 - W#16#7FFF	The first subslot number for the submodule is one. The last subslot number for the submodule is W#16#7FFF.

W#16#8000 - W#16#8FFF	Used for 16 interface modules with up to 255 ports
	0x8IPP with I counting interfaces and P counting ports;
	PP := 1 to 255;
	I := 0 to 15 if PP=00; describes the actual interface module
	for example, 8001: I=0 and PP=01, port 1 of interface 0

W#16#9000 - W#16#FFFF	Reserved

2.5.16 USI

Table 2-43 USI (UserStructureIdentifier) coding

Value (hexadecimal)	Meaning	Description
W#16#0000 - W#16#7FFF	Vendor-specific	When combined with the interrupt type, diagnostics incoming/outgoing is vendor-specific Diagnostics in AlarmNotification and diagnostics data. Vendor-specific usage in combination with other interrupt types.
W#16#8000	ChannelDiagnosis	Used only in combination with ChannelDiagnosis in AlarmNotification and diagnostics data.
W#16#8001	Multiple	Only used in combination with data which correspond to the structure of “(BlockHeader, Data*)*”. BlockType always corresponds to the AlarmType used.
W#16#8002	ExtChannelDiagnosis Data	Used only in combination with ChannelDiagnosisWithAddInfo in AlarmNotification and diagnosis data.
W#16#8003	Qualified	Graded extended channel diagnostics
W#16#8004 - W#16#80FF	Reserved	
W#16#8100	Maintenance	Maintenance
W#16#8101 - W#16#8FFF	Reserved	
W#16#9000 - W#16#9FFF	Reserved for profiles	Reserved for profiles
W#16#A000 - W#16#FFFF	Reserved	

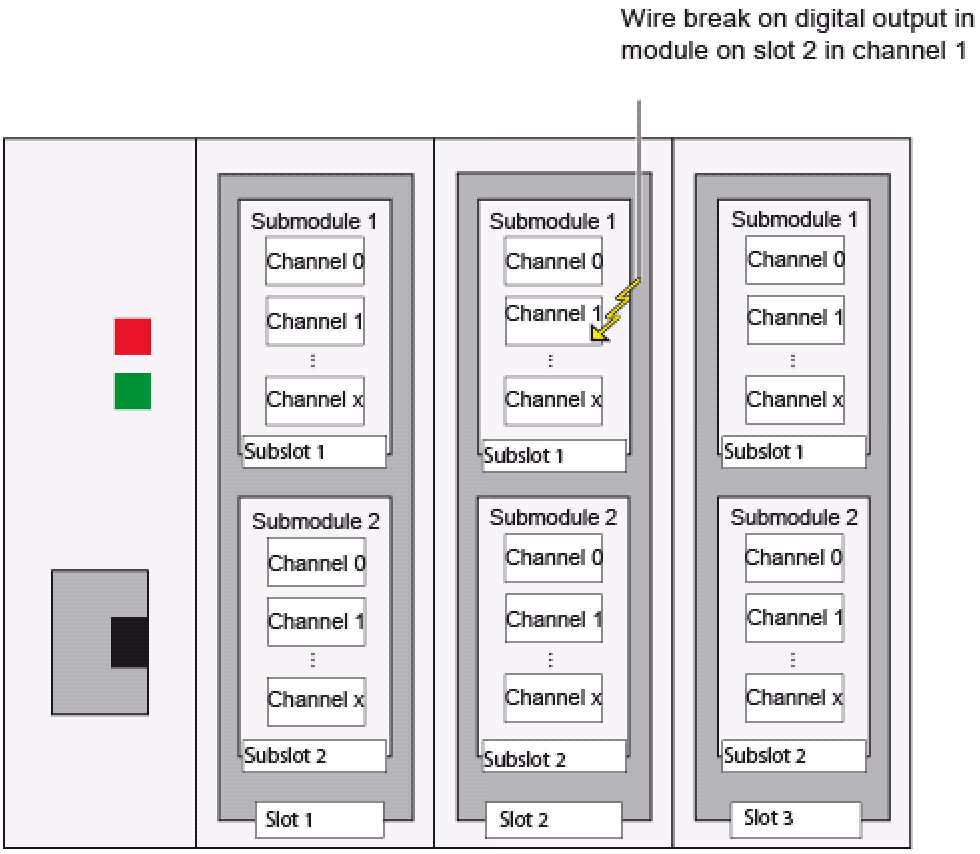
3 Examples of diagnostics data records

3.1 Example of diagnostics data record W#16#800A

Example of diagnostics record W#16#800A

Record W#16#800A is read from subslot 1 / slot 2. Wire break at one of the two submodule channels (outputs).

Figure 3-1 Wire break at channel 1 / subslot 1 / slot 2



The content of the diagnostics record is then as follows:

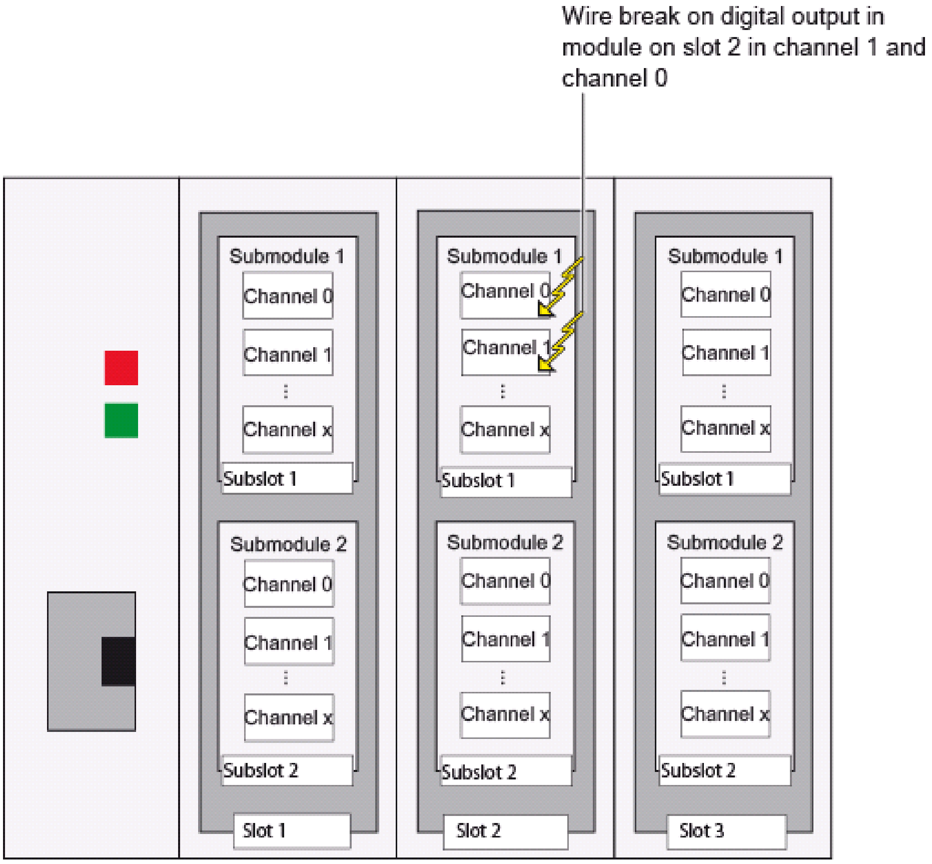
Table 3-1 Example of diagnostic record W#16#800A for a faulty channel

DB name	Content	Comment
Diagnostics record W#16#800A returns one record for a subslot (subslot level was addressed with this record); only one channel diagnostics record is available as only one channel is faulty.		
BlockHeader	BlockHeader consists of BlockType, BlockLength, BlockVersion	
BlockType	W#16#0010	DiagnosisBlock, that is, this is a diagnostics record
BlockLength	W#16#0016	Decimal 22, that is, 22 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
API	DW#16#00000000	API is 0; that is, it has no profile
Slotnumber	W#16#0002	Module in slot 2
SubslotNumber	W#16#0001	First subslot
ChannelNumber	W#16#8000	Diagnostics at submodule level
ChannelProperties	W#16#0800	In the binary system: 0000 1000 0000 0000 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelProperties.Type (Bit 0 - 7)	W#16#00	Set to 0 if ChannelNumber = W#16#8000
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	000	Vendor-specific
USI	W#16#8000	Three data blocks are appended to this DB for each faulty channel: ChannelNumber, ChannelProperties and ChannelErrorType
The data blocks shown below are generated for each faulty channel; channel 1 in this example		

DB name	Content	Comment
ChannelNumber	W#16#0001	Channel 1
ChannelProperties	W#16#4801	In the binary system: 0100 1000 0000 0001 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelPropertiesType (Bit 0 - 7)	B#16#01	1 bit
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	010	Output
ChannelErrorType	W#16#0006	Wire break

Record W#16#800A is read from slot 2 / subslot 1. Wire break at one of the two submodule channels (outputs).

Figure 3-2 Wire break at the digital outputs channel 0 & channel 1 / sublot 1 / slot 2



The content of the diagnostics record is then as follows:

Table 3-2 Example of diagnostics record W#16#800A for two faulty channels

DB name	Content	Comment
Diagnostics record W#16#800A returns just one record for the subslot (subslot level was addressed with this record); two records are returned because there are two faulty channels.		
BlockHeader	BlockHeader consists of BlockType, BlockLength, BlockVersion	
BlockType	W#16#0010	DiagnosisBlock, that is, this is a diagnostics record
BlockLength	W#16#001C	Decimal 28, that is, 28 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
API	DW#16#00000000	API is 0; that is, it has no profile
Slotnumber	W#16#0002	Module in slot 2
SubslotNumber	W#16#0001	First subslot
ChannelNumber	W#16#8000	Diagnostics at submodule level
ChannelProperties	W#16#0800	In the binary system: 0000 1000 0000 0000 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelProperties. Type (Bit 0 - 7)	B#16#00	Set to 0 if ChannelNumber = W#16#8000
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction **(Bit 13-15)**	000	Vendor-specific
USI	W#16#8000	Three data blocks are appended to this DB for each faulty channel: ChannelNumber, ChannelProperties and ChannelErrorType
Followed by channel diagnostics for faulty channel 1		
ChannelNumber	W#16#0001	Channel 1
ChannelProperties	W#16#4801	In the binary system: 0100 1000 0000 0001 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelPropertiesType (Bit 0 - 7)	B#16#01	1 bit

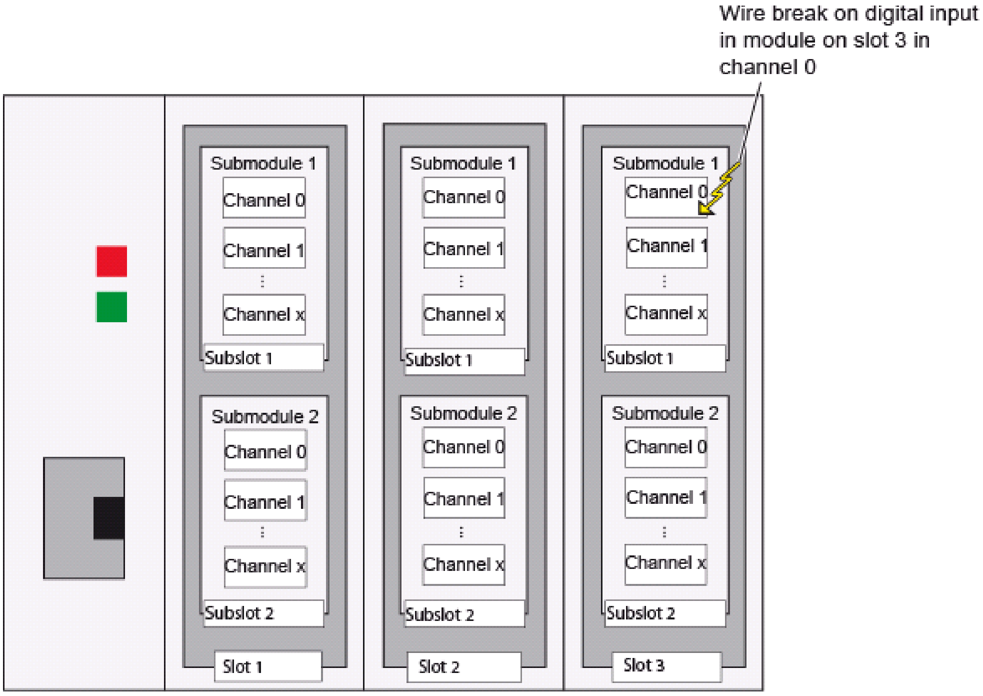
DB name	Content	Comment
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	010	Output
ChannelErrorType	W#16#00 06	Wire break
Followed by channel diagnostics for faulty channel 0		
ChannelNumber	W#16#0000	Channel 0
ChannelProperties	DW#16#4801	In the binary system: 0100 1000 0000 0001 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelPropertiesType (Bit 0 - 7)	B#16#01	1 bit
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	010	Output
ChannelErrorType	W#16#0006	Wire break

3.2 Example of the diagnostics data record W#16#800C

Example of diagnostics record W#16#800C

Diagnostics record W#16#800C is read from subslot 1 / slot 3. Short-circuit at one of the two submodule channels (inputs).

Figure 3-3 Short circuit at the digital input channel 0 / sublot 1 / slot 3



The content of the diagnostics record is then as follows:

Table 3-3 Example of diagnosis record W#16#800C for one faulty channel

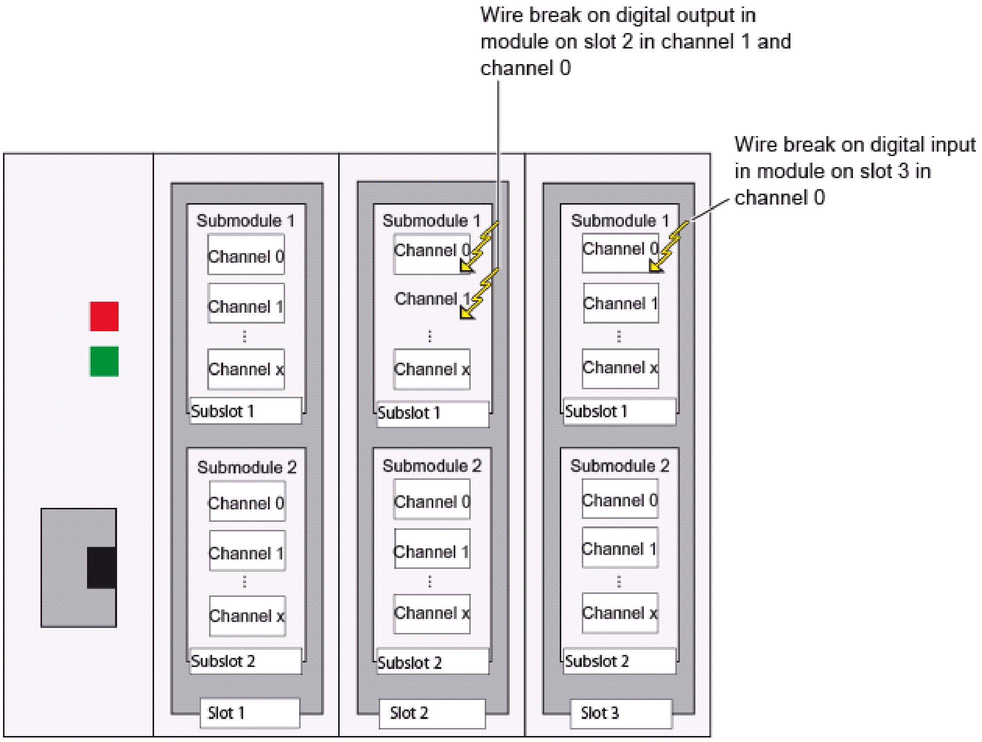
DB name		Content	Comment
Diagnostics record W#16#800C returns one record for a subslot (subslot level was addressed with this record); it returns one channel diagnostics record because only one channel is faulty.			
BlockHeader		BlockHeader consists of BlockType, BlockLength, BlockVersion	
	BlockType	W#16#0010	DiagnosisBlock, that is, this is a diagnostics record
	BlockLength	W#16#0016	Decimal 22, that is, 22 bytes are appended to the BlockLength data block.
	BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
API		DW#16#00000000	API is 0; that is, it has no profile
Slotnumber		W#16#0003	Module in slot 3
SubslotNumber		W#16#0001	First subslot
ChannelNumber		W#16#8000	Diagnostics at submodule level
ChannelProperties		W#16#0800	In the binary system: 0000 1000 0000 0000

DB name	Content	Comment
		ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelProperties.Type (Bit 0 - 7)	B#16#00	Set to 0 if ChannelNumber = W#16#8000
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	000	Vendor-specific
USI	W#16#8000	Three data blocks are appended to this DB for each faulty channel: ChannelNumber, ChannelProperties and ChannelErrorType
Followed by channel diagnostics for faulty channel 0		
ChannelNumber	W#16#0000	Channel 0
ChannelProperties	W#16#2801	In the binary system: 0010 1000 0000 0001 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelPropertiesType (Bit 0 - 7)	B#16#01	1 bit
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier** **(Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction **(Bit 13-15)**	001	Input
ChannelErrorType	W#16#0001	Short-circuit

3.3 Example of diagnostics data record W#16#E00C

Diagnostics record W#16#E00C is read from a device (AR). Submodule 1 (outputs) of slot 2 returns two wire break diagnostics records, and submodule 1 (inputs) of slot 3 reports a short-circuit.

Figure 3-4 Short-circuit at digital input channel 0 / sublot 1 / slot 3; wire break at digital output channel 0 & channel 1 / subslot 1 / slot 2


Wire break on digital output in module on slot 2 in channel 1 and channel 0
Wire break on digital input in module on slot 3 in channel 0
Submodule 1
Channel 0
Channel 1
Channel x
Subslot 1
Submodule 2
Channel 0
Channel 1
Channel x
Subslot 2
Slot 1
Slot 2
Slot 3

This gives the following diagnostics record:

Table 3-4 Example of diagnostics record W#16#E00C with two diagnostics records

DB name	Content	Comment
This is followed by a record for slot 2 which contains the faulty channels 1 and 0		
BlockHeader	BlockHeader consists of BlockType, BlockLength, BlockVersion	
BlockType	W#16#0010	DiagnosisBlock, that is, this is a diagnostics record
BlockLength	W#16#001C	Decimal 28, that is, 28 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
API	DW#16#00000000	API is 0; that is, it has no profile
Slotnumber	W#16#0002	Module in slot 2
SubslotNumber	W#16#0001	First subslot
ChannelNumber	W#16#8000	Diagnostics at submodule level
ChannelProperties	W#16#0800	In the binary system: 0000 1000 0000 0000 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelProperties.Type (Bit 0 - 7)	B#16#00	Set to 0 if ChannelNumber = W#16#8000 #16#8000 ist
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	000	Vendor-specific
USI	W#16#8000	Three data blocks are appended to this DB for each faulty channel: ChannelNumber, ChannelProperties and ChannelErrorType
Followed by channel diagnostics for faulty channel 1		

DB name	Content	Comment
ChannelNumber	W#16#0001	Kanal 1
ChannelProperties	W#16#4801	Im Dualzahlensystem: 0100 1000 0000 0001 ChannelProperties besteht aus (Bit 0-7) ChannelProperties.Type, (Bit 8) ChannelProperties.Accumulativ, (Bit 9) MaintenanceRequired, (Bit 10) MaintenanceDemanded, (Bit 11 - 12) ChannelProperties.Specifier (Bit 13 - 15) ChannelProperties.Direction
ChannelProperties.Type (Bit 0 - 7)	B#16#01	1 bit
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier** **(Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction **(Bit 13-15)**	010	Output
ChannelErrorType	W#16#0006	Wire break
Followed by channel diagnostics for faulty channel 0		
ChannelNumber	W#16#0000	Channel 0
ChannelProperties	W#16#4801	In the binary system: 0100 1000 0000 0001 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelPropertiesType (Bit 0 - 7)	B#16#01	1 bit
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	010	Output
ChannelErrorType	W#16#0006	Wire break
This is followed by a record for slot 3 which contains the faulty channel 0		

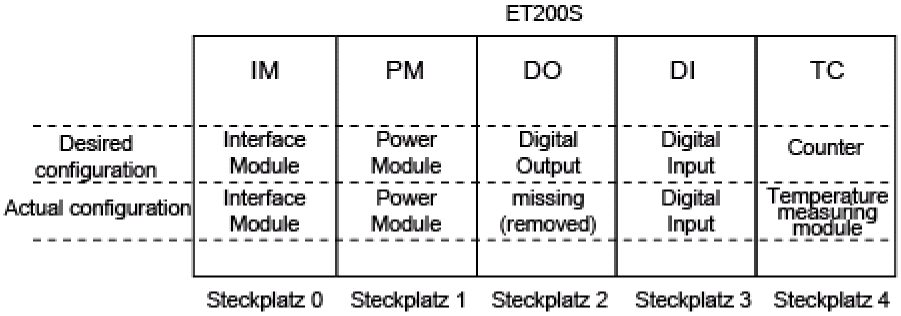
DB name	Content	Comment
BlockHeader	BlockHeader consists of BlockType, BlockLength, BlockVersion	
BlockType	W#16#0010	DiagnosisBlock, that is, this is a diagnostics record
BlockLength	W#16#0016	Decimal 22, that is, 22 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
API	DW#16#00000000	API is 0; that is, it has no profile
Slotnumber	W#16#0003	Module in slot 3
SubslotNumber	W#16#0001	First subslot
ChannelNumber	W#16#8000	Diagnostics at submodule level
ChannelProperties	W#16#0800	In the binary system: 0000 1000 0000 0000 ChannelProperties consists of (bit 0 to 7) ChannelProperties.Type (bit 8) ChannelProperties.Accumulative (bit 9) MaintenanceRequired (bit 10) MaintenanceDemanded (bit 11 to 12) ChannelProperties.Specifier (bit 13 to 15) ChannelProperties.Direction
ChannelProperties.Type (Bit 0 - 7)	B#16#00	Set to 0 if ChannelNumber = W#16#8000
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	000	Vendor-specific
USI	W#16#8000	This DB is appended three data blocks: ChannelNumber, ChannelProperties and ChannelErrorType
Followed by channel diagnostics for faulty channel 0		

DB name	Content	Comment
ChannelNumber	W#16#0000	Channel 0
ChannelProperties	W#16#2801	Im Dualzahlensystem: 0010 1000 0000 0001 ChannelProperties besteht aus (Bit 0-7) ChannelProperties.Type, (Bit 8) ChannelProperties.Accumulativ, (Bit 9) MaintenanceRequired, (Bit 10) MaintenanceDemanded, (Bit 11 - 12) ChannelProperties.Specifier (Bit 13 - 15) ChannelProperties.Direction
ChannelPropertiesType (Bit 0 - 7)	B#16#01	1 bit
ChannelProperties. Accumulativ (Bit 8)	0	No channel error group message
MaintenanceRequired (Bit 9) **MaintenanceDemanded (Bit 10)** **ChannelProperties.Specifier (Bit 11 - 12)**	0100	Diagnostics pending
ChannelProperties.Direction (Bit 13-15)	001	Input
ChannelErrorType	W#16#0001	Short-circuit

3.4 Example of the configuration data record W#16#E000

Configuration record W#16#E000 contains the target configuration of an IO device. This example shows the configuration of 5 slots on an IO device.

Figure 3-5 Configuration error in configuration records W#16#E000, W#16#E001 and W#16#E002



This gives the following configuration record:

Table 3-5 Example of a configuration record W#16#E000

DB name	Content	Comment
One configuration record is generated for each AR		
BlockHeader	BlockHeader besteht aus BlockType, BlockLength, BlockVersion	
BlockType	W#16#0012	ExpectedIdentificationDataBlock, that is, this is the diagnostics record for the expected configuration
BlockLength	W#16#0050	Decimal 80, that is, 80 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
NumberOfAPIs	W#16#0001	An API is available
Data blocks generated for each API:		
API	DW#16#00000000	API=0, that is, no profile available
NumberOfSlots	W#16#0005	Five slots are configured for this device
Data blocks generated for each configured slot: The next five data blocks return information about slot 0		
SlotNumber	W#16#0000	Slot number = 0
ModuleIdentNumber	DW#16#00000322	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Slot number = 1
SubmoduleIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 1		

DB name	Content	Comment
SlotNumber	W#16#0001	Slot number = 1
ModuleIdent Number	DW#16#00000684	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Slot number = 1
SubmoduleIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 2		
SlotNumber	W#16#0002	Slot number = 2
ModuleIdentNumber	DW#16#000088a1	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Slot number = 1
SubmoduleIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 3		
SlotNumber	W#16#0003	Slot number = 3
ModuleIdentNumber	DW#16#00001094	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Slot number = 1
SubmoduleIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 4		
SlotNumber	W#16#0004	Slot number = 4
ModuleIdentNumber	DW#16#0000d6d8	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Slot number = 1
SubmoduleIdentNumber	DW#16#00000000	Vendor-specific

3.5 Example of the configuration data record W#16#E001

Example of a configuration record W#16#E001

Configuration record W#16#E001 contains the actual configuration of an IO device. This example shows four slots, as the module has been removed from slot 2 (target configuration corresponds to five slots; see section 3.4).

Table 3-6 Example of a configuration record W#16#E001

DB name	Content	Comment
One configuration record is generated for each AR		
BlockHeader	BlockHeader consists of BlockType, BlockLength, BlockVersion	
BlockType	W#16#0013	RealIdentificationData, that is, this record returns the actual configuration
BlockLength	W#16#0042	Decimal 66, that is, 66 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0101	BlockVersion 0101 is assigned to this diagnostics record
NumberOfAPIs	W#16#0001	An API is available
Data blocks generated for each API:		
API	DW#16#00000000	API=0, that is, no profile available
NumberOfSlots	W#16#0004	Four modules are physically available
Data blocks generated for each existing module: The next five data blocks return information about slot 0		
SlotNumber	W#16#0000	Slot number = 0
ModuleIdentNumber	DW#16#00000322	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Subslot number = 1
SubmodulIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 1		
SlotNumber	W#16#0001	Slot number = 1
ModuleIdentNumber	DW#16#00000684	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Subslot number = 1
SubmodulIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 3		
SlotNumber	W#16#0003	Slot number = 3
ModuleIdentNumber	DW#16#00001094	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		

DB name	Content	Comment
SubslotNumber	W#16#0001	Subslot number = 1
SubmodulIdentNumber	DW#16#00000000	Vendor-specific
The next five data blocks return information about slot 4		
SlotNumber	W#16#0004	Slot number = 4
ModuleIdentNumber	DW#16#000017FF	Vendor-specific
NumberOfSubslots	W#16#0001	One configured submodule
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Subslot number = 1
SubmodulIdentNumber	DW#16#00000000	Vendor-specific

3.6 Example of the configurationdata record W#16#E002

Example of a configuration record W#16#E002

Configuration record W#16#E002 returns the difference between the target and actual configuration (see sections 6.1 and 6.2) of an IO device. The module is missing from slot 2, and an incorrect module has been inserted in slot 4.

Table 3-7 Example of a diagnostics record W#16#E002

DB name	Content	Comment
This configuration record returns the difference between the target and actual configuration		
BlockHeader	BlockHeader consists of BlockType, BlockLength, BlockVersion	
BlockType	W#16#8104	ModuleDiffBlock, that is, this record returns the difference between the configured and diagnosed modules
BlockLength	W#16#0026	Decimal 38, that is, 38 bytes are appended to the BlockLength data block.
BlockVersion	W#16#0100	BlockVersion 0100 is assigned to this diagnostics record
NumberOfAPIs	W#16#0001	An API is available
Data blocks generated for each API:		
API	DW#16#00000000	API=0, that is, no profile available
NumberOfModules	W#16#0002	Two modules differ from the target configuration
Data for incorrect module		
SlotNumber	W#16#0002	Module in slot 2
ModuleIdentNumber	DW#16#000088a1	ModuleIdentNumber of the missing module
ModuleState	W#16#0000	Module missing
NumberOfSubslots	W#16#0000	No submodule available as the module is not inserted
Data for incorrect module		

DB name	Content	Comment
SlotNumber	W#16#0004	Module in slot 4
ModuleIdentNumber	DW#16#000017ff	ModuleIdentNumber of the incorrect module
ModuleState	W#16#0001	Incorrect module
NumberOfSubslots	W#16#0001	A submodule is available
Data blocks generated for each configured subslot:		
SubslotNumber	W#16#0001	Submodule at subslot 1
SubmoduleIdentNumber	DW#16#00000000	Vendor-specific
SubmoduleState	W#16#9000 In the binary system: 1001 0000 0000 0000	
	Bit 15 = 1	Format indicator is 1
	Bit 11-14 = 0010	Incorrect module
	Bit 7-10 = 0000	AR has a submodule
	Bit 6 = 0	No diagnostic available
	Bit 5 = 0	No MaintenanceDemanded
	Bit 4 = 0	No MaintenanceRequest
	Bit 3 = 0	No graded extended channel diagnostics
	Bit 0-2 = 000	No meaning

4 Related Literature

Internet Link Specifications

This list is not complete and only represents a selection of relevant information.

Table 4-1

	Subject	Title
\1\	Reference to the entry	http://support.automation.siemens.com/WW/view/en/24000238
\2\	Siemens I IA/DT Customer Support	http://support.automation.siemens.com
\3\	Homepage of the PROFIBUS User Organisation	http://www.profibus.com
\4\	System Software for S7-300/400 System- and Standard functions	http://support.automation.siemens.com/WW/view/en/44240604
\5\	PROFINET system description	http://support.automation.siemens.com/WW/view/en/19292127

5 History

Table 5-1

Version	Date	Modifications
V1.0	07.10.2011	Extract from the manual „From PROFIBUS DP to PROFINET IO““