Profile	: Encoders
Profile Number	: 71
Date	: March 17, 1995
Published by	: ENCOM User Group
	Postfach 11 03, D-32817 Blomberg
	Tel.: *49 - 52 35 - 34 15 84
	Fax: *49 - 52 35 - 34 18 62
Order No.	: 71

Contents

1. Preface

Within the framework of production automation, increasingly efficient and flexible systems are required in the field of industrial sensors and actuators. Encoders can meet these requirements. Open and standardized communication possibilities are, however, a prerequisite for their full integration into complex manufacturing schedules.

The idea underlying open systems is to enable information exchange between application functions that are implemented on devices from various manufacturers.

Belonging to these are fixed application functions, a uniform application interface for communication and a uniform transmission medium.

The ENCOM User Group has set itself the task of standardizing the most important decoder device functions and gathering them into this profile for various encoders.

In order to define the encoder device functions independently of the communication medium, an internationally recognized and standardized application interface (DIN 19245 Teil 2) for communication has been used. In this way, consistency with MSS was achieved.

InterBus-S system, which fulfills the requirements of the sensors and actuators with respect to real-time behavior and standardized application interfaces, was selected as transmission medium.

This profile for encoders is directed at users and device manufacturers of encoders that are to be operated on a sensor/actuator bus.

This profile definition is a useful addition to standardized communication for the user and provides a generally valid agreement concerning data content and device behavior. These function settings standardize several important encoder device parameters. In this way, when using these standard parameter, devices from various manufacturers behave in the same way in connection with the communication medium.

An independent committee of experts will be appointed for conformance testing and certification of products with the encoder profile.

Additions will be made in the course of continued standardization.

2. Definitions

Device Profile

The device profile determines the visible application functions for communication. The application functions are mapped onto the communication by the following definitions:

- by the communication profile,
- by the interaction between the application functions inasmuch as they are executed via the communication system, as well as
- by the communication services used and the communication objects that can thereby be manipulated.

The result of this mapping is the visible behavior of the application. The definitions of one application profile allow the inter-operability in one application area when the device characteristics used allow this.

Characteristics of the device that are significant for the user are also defined.

A differentiation is made between mandatory functions, optional and manufacturer-specific device function, and parameters.

If the user restricts himself to mandatory functions or parameters, the device is interchangeable when the device characteristics and device settings allow this. Regarding the communication, devices with identical parameters are always interchangeable, independent of the function.

Communication Profile

In the communication profile, the degrees of freedom contained in the specification of the transmission medium are limited or classified specific to the application or device group. Communication services and parameters that are marked as optional in the specification are determined in the communication profile.

All optional functions and parameters that are not named in the communication profile remain optional. Mandatory services and parameters also remain mandatory when they are not named in the profile.

Furthermore, value ranges of attributes and parameters are limited or determined in the profile.

InterBus-S serves as communication medium.

Sensor/Actuator

This profile contains the basic functions that all sensor and actuator devices must provide to the user. These are mainly communication functions and device information.

Communication Interface

The communication interface consists of a channel for services compatible to DIN 19 245 Part 2 (Peripherals Communication Protocol Channel, PCP-Channel) and a channel for process data.

All communication objects can be accessed via the PCP channel (only with class 4, see "Classification of Encoders"). Access is via confirmed services, i.e. access to a communication object is acknowledged by the encoder. The process data channel is used for fast transmission of particular communication objects. It allows acknowledged access (class 3) and unconfirmed access (class 1 - 4) to the communication objects.

In the following, the data that is transmitted via the process data channel is referred to as process data.

Data is transmitted unconfirmed and equidistant via the process data channel. It is up to 16 bytes wide. Each byte can be both read and written.

The direction of the process data is seen from the bus, i.e.

- process output data is data that is read by the encoder from the process data channel.
- process input data is data that is written into the process data channel by the encoder.

Encoders

Encoders are angle-of-rotation transducers with coded or incremental detection of the measured quantity. These angle-of-rotation transducers are known as ’absolute or incremental angle encoders’, ’absolute rotary transducers and ’rotary position transducers’.

Step Number

The step number is the physical position of the encoder related to its absolute zero point of the encoder disk.

Measuring steps

A measuring step is the smallest measurable angular change by the encoder disk division.

Step

A step is a measuring step that has been changed in quantity by parameterization.

Index, Sub-Index

The index is used to address a parameter (communication object). The sub-index addresses a sub-parameter (an element of a communication object) within a parameter that is stored as a structure.

Substitute Values

When the optional communication objects are not implemented, the device behaves according to the substitute value defined for this parameter.

Mandatory Range

The mandatory range is the value range within which a parameter can definitely be parameterized, as long as it is implemented.

State Machine

In this profile, several functions are described with the aid of a state machine. A state represents a particular internal and external behavior. It can only be left via defined events. The events are assigned corresponding state transitions. In a transition, operations can be carried out. The state behavior is changed during the transition. When the transition is completed, the current state is changed to the following state.

Definition of the Functional Units

The device function is described by a function unit (see Figure 1). The function is controlled and parameterized by the inputs. Furthermore, internal signals can effect the function or the function is influenced by an internal parameter. The output of the function can be switched to the inputs of other functions or made accessible via the bus.

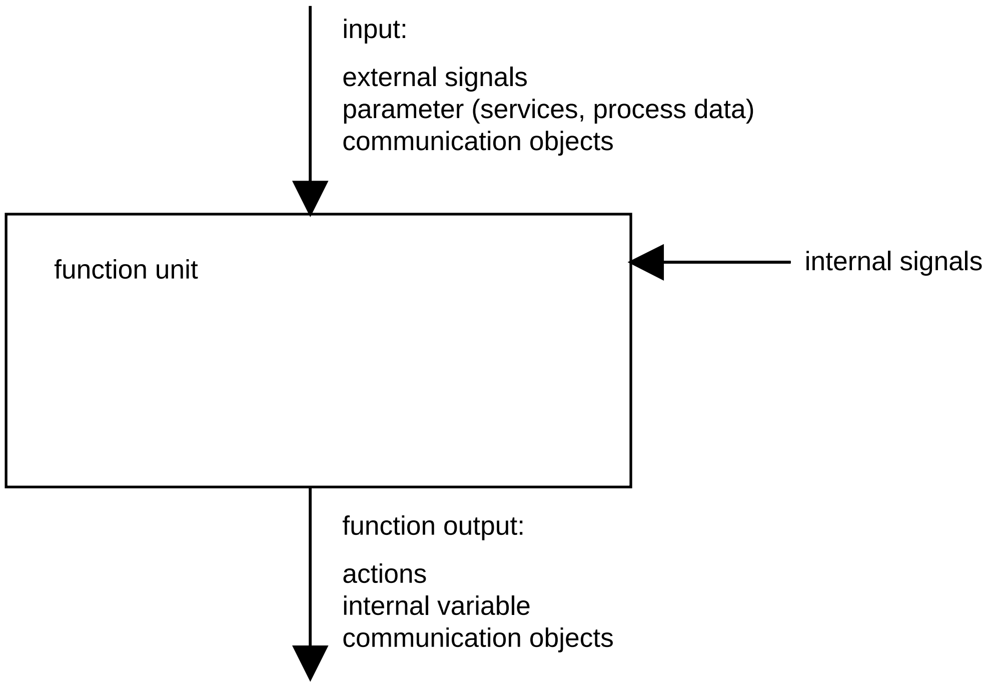


Figure 1: Function unit

2.1. References

The application protocol and the data structure of the class 4 encoder correspond to InterBus-S Club Guideline.

The application interface for communication via InterBus-S parameter channel corresponds to InterBus-S Club guidelines.

Determinations for data transmission via the process data channel are based on InterBus-S Club guidelines and the DIN 19258 Draft Recommendation.

The terms used correspond to the DIN 32 878 Draft Recommendation.

Authors:

Mr. Wiemann	FRABA, Cologne
Mr. Schneider	Gelma, Bonn
Mr. Krumsiek	Phoenix Contact, Blomberg
Mr. Schulz	TWK, Düsseldorf
Mr. Henke	TWK, Hamburg

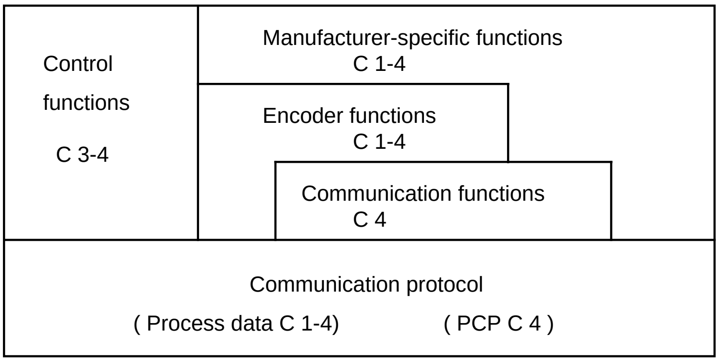
2.2. Abbreviations

m	Mandatory
MAP	Manufacturing Automation Protocol
MMS	Manufacturing Messages Services
o	Optional
OD	Object dictionary
PCP	Peripherals Communication Protocol
PD	Process data
PMS	Peripherals Message Specification
VFD	Virtual Field Device
.con	Confirmation primitive
.ind	Indication primitive
.req	Request primitive
.res	Response primitive
ID-Code	identification code

3. Summary

3.1. Function Structure of Encoders

This profile defines application functions of encoders. The application functions are divided into device functions, communication functions and control functions (as shown in Figure 2). Furthermore, free spaces are defined for the manufacturer-specific functions.



C = profile class

Figure 2: Application functions of encoders

Every application function is described with the aid of a function block. The device behavior is described by a state machine in the control functions. This profile takes into account that a separate hardware can be available for the communication and for tasks connected to communication.

The encoder functions consist of the standardized functions that are addressed via the communication functions determined in the Sensor/Actuator Profile.

The control functions serve to coordinate the function ranges, whereby a further division is possible into standardized and manufacturer-specific functions. The device control is described in this profile as standardized control function.

The manufacturer-specific functions can also be addressed via the communication functions.

4. Device Characterization

The connection from the automation equipment (PLC = programmable logic controller) to the process is made by an encoder. Simply put, the encoder converts physical values from the process into electrical signals for the automation equipment. Encoders can be active or passive devices on the bus.

The market for general encoders requires a wide spectrum of different devices with regard to function and price. Due to the open structure of the encoder profile, the most varied functions are covered.

The device functions and parameters are differentiated as mandatory, optional, and manufacturer-specific. If the user limits himself to the mandatory functions or mandatory parameters, the encoders ccan be interchangeable.

Regarding the communication, devices with identical parameters are always interchangeable, independent of the function.

4.1. Encoder Classification

Four device classes are defined for encoders in this profile:

Profile Class 1

The simplest encoders that only deliver 16 bit process data are gathered in class 1. The position actual value (output value) of the encoder is binary coded and right aligned in the 16 bit process data word.

Profile Class 2

The simplest encoders that only deliver 32 bit process data are gathered in class 2. The position actual value of the encoder is binary coded and right aligned in the 32 bit process data word.

Profile Class 3

Encoders that deliver 32 bit process data are gathered in class 3. These 32 bits contain a 25-bit position actual value and a 7 bit status bit and control bits. The position actual value of the encoder is coded acc. to the specifications of the manufacturer and right aligned in the bits 0 - 24 of the process data word. Bits 25 - 25 contain the status bit and control bits. They are fixed in this profile class. When all control bits are set to 0 and the OPERATION condition is displayed on the status bits, the encoder outputs a valid value for bits 0 to 24.

b31 b25	b24 b0
Control/Status	Position actual value

Profile Class 4

- 32 bit binary process data, right aligned
  4 control/status bits
- PCP channel

Encoders that deliver 32 bit process data are gathered in class 4. These 32 bits contain a 28 bit output value and a 4 bit status bit and control bits. The position actual value of the encoder is binary coded and right aligned in the bits 0 - 27 of the process data word. Bits 28 - 31 contain the status bit and control bits. The status bits and control bits are defined in this profile class. Furthermore, the parameter channel is compulsory in this profile class.

b31 b28	b27 b0
Control/Status	Position actual value

4.2. Device Data

Encoders are characterized by the fact that changes in the ’position’ process variable position are caused by analog or digital signals. Additional parameters are required to adapt the encoders optimally to the process. In addition to the actual value information as such, the devices also provide, e.g. the values of the steps that are provided as actual value.

An analysis of the parameters named leads to the formation of two data classes: the group of data to be cyclically transmitted (e.g., the position actual value) and the group of acyclic data transmission (e.g., the resolution of the position values). This data classification is not only to be found in encoder technology, but in the whole field of sensors/actuators.

When considering the quality of both data classes, there are significant differences regarding the time requirements and the data width. The cyclical data generally carries low amounts of data of 16 - 32 bits and has to be transmitted at a cycle time of some milliseconds. The group of acyclical data, in contrast, carries considerably higher amounts of data, e.g. 20 * 16 bit information units. This data is, in contrast to that of the cyclical data, seldom changed. The time requirement, the cycle time, is oriented to that of the cyclical data.

5. Application and Device Characteristics

In this chapter, the entire application is described from the point of view of communication. The application is divided into the function blocks shown in Figure 3:

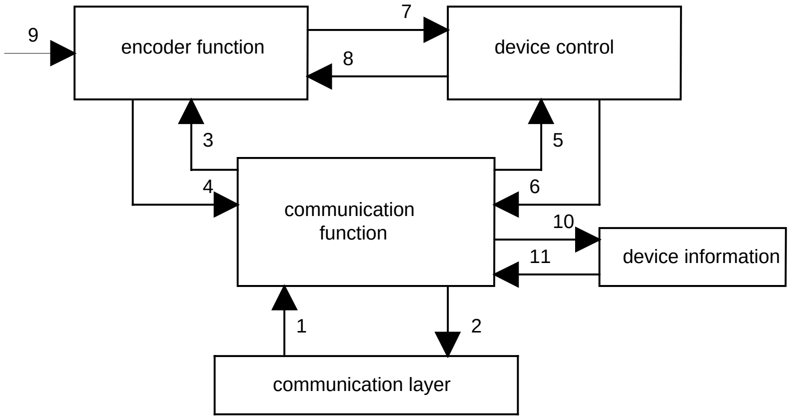


Figure 3: Function Blocks of an Application

Communication Function

The communication function executes all communication-specific functions.

Device Control

The device control function block assumes control of the entire device functions.

Encoder Function

The encoder function executes all encoder-specific functions.

Device Information

The device information administers information on the device in a non-volatile memory module.

Communication Layer

The communication layer contains a DIN 19245 Part 2 corresponding to layer 7 and a DIN 19258 Draft Recommendation corresponding to layer 7.

Interaction between the Function Blocks

1	Data from the bus system	7	Status of the encoder function, malfunction
2	Data to the bus system	8	Control of the encoder function
3	Setting of encoder parameters	9	Physical position
4	Read-out of encoder parameters	10	Storing of device information
5	Commands to the device control (control word)	11	Read-out of device information
6	State of the device control		

5.1. Device Control

The device control function block (see Figure 4) assumes control of the entire device function. The control sequence is described by a state machine. The device control is influenced by the control word, by internal signals, and by malfunctions. The device control effects the encoder function. The status word is generated from the device state and internal signals and can be read out from the bus.

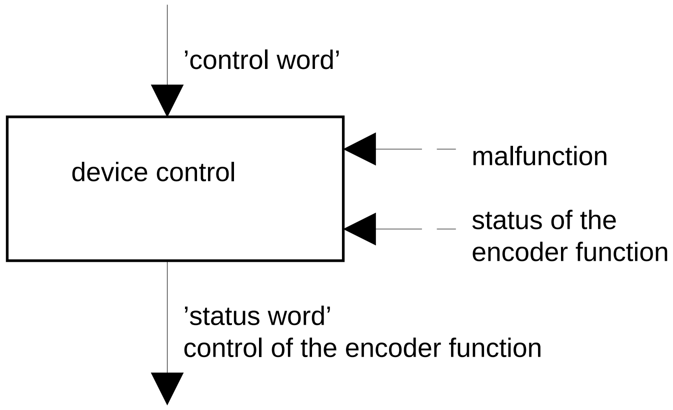


Figure 4: Device control function block

’Control Word’

The encoder can be controlled by the bits in the ’control word’.

’Status Word’

The ’status word’ parameter shows information concerning the status and messages of the encoder.

5.1.1. Device Control State Machine

The state machine (see Figure 5) describes the device states and the possible control sequence of the encoder. A state represents a particular internal and external behavior. The state can be switched-over and a control flow thereby executed with device control commands and internal events. The current state can be read-out by the status word.

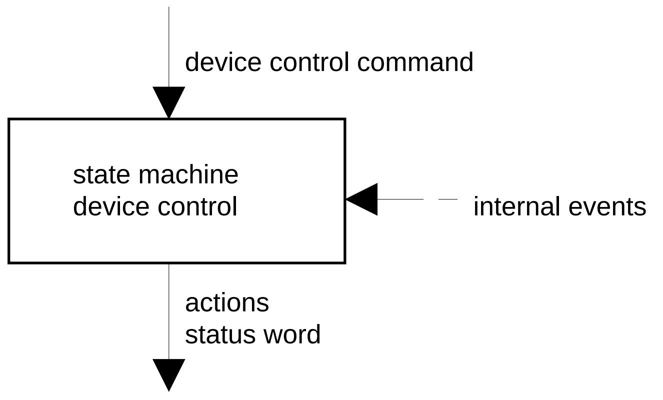


Figure 5: 'Device control' state machine

State Diagram (see Figure 6)

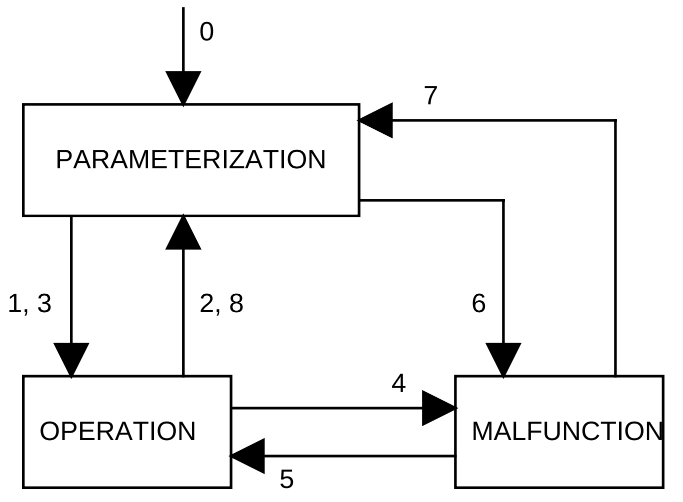


Figure 6: Device control state diagram

The device control commands result in the following transitions in the state machine:

Device control command :	State transitions triggered:
Enable operation	3, 5

Device Control States

OPERATION
in OPERATION state, the position actual value (output code) is output on the process data channel.

PARAMETERIZATION
In PARAMETERIZATION state, the parameters are changed that have an influence on the value transmitted via the process data. In this state, the position actual value is not output on the process data channel.

MALFUNCTION
This state is assumed when a malfunction has occurred. The reason for the malfunction is stored in the malfunction code parameter and in the status bit. In this state, the last valid position actual value is output.

State Transitions of the Device Control

0 Power Up --> PARAMETERIZATION
Event : Power On
Action : none

1 PARAMETERIZATION --> OPERATION
Event : initialization completed
Action : output position actual value

2 OPERATION --> PARAMETERIZATION
Event : description of a parameter that influences the process data output value.
Action : none

3 PARAMETERIZATION --> OPERATION
Event : control command ’enable operation’
Action: output position actual value
The parameters have taken on the parameter values

4 OPERATION --> MALFUNCTION (optional state transition)
Event : malfunction in encoder
Action : freeze position actual value
generate malfunction code and output on status bits

5 MALFUNCTION --> OPERATION
Event : control command ’enable operation’
Action : output position actual value

6 PARAMETERIZATION --> MALFUNCTION
Event : control command ’enable operation’
and
parameter inconsistent or invalid values
Action : generate malfunction code and output on status bit

7 MALFUNCTION --> PARAMETERIZATION
Event : description of a parameter that influences the process data output value.
control command ’parameterization’
Action : none

8 OPERATION --> PARAMETERIZATION (Profile Class 3)
Event : control command ’parameterization’
Action : none

The state is only changed once the actions have been entirely executed. The order of the actions corresponds with their processing during the state change. After complete processing of the actions, the next state is reached and new commands are accepted.

5.1.2. Control Word

The 'control word' and the internal signals result, by logical operation, in the device control commands, which effect the state machine of the device control. In this way, functions are triggered and operating states of the device are determined. The control word comprises 16 bits with the following designations:

Control Word for Profile Class 3

Bit	Name	Mandatory	Bit No. in PD Channel
0	Reserved		
1	Reserved		
2	Reserved		
3	Reserved		
4	Reserved		
5	Reserved		
6	Reserved		
7	Reserved		
8	Reserved		
9	Parameter no.	X	25
10	Parameter no.	X	26
11	Parameter no.	X	27
12	Parameter no.	X	28
13	Manufacturer-specific		29
14	Set zero shift	X	30
15	Enable operation	X	31

Control Word for Profile Class 4

Bit	Name	Mandatory
0	Manufacturer-specific	
1	Manufacturer-specific	
2	Manufacturer-specific	
3	Manufacturer-specific	
4	Manufacturer-specific	
5	Manufacturer-specific	
6	Manufacturer-specific	
7	Manufacturer-specific	
8	Manufacturer-specific	
9	Reserved	X
10	Reserved	X
11	Reserved	X
12	Reserved	X
13	Manufacturer-specific	
14	Set zero shift	X
15	Enable operation	X

Device Control Commands

The device control commands are triggered by the following bit combinations in the control word:

	Control Word (Bit)							
	Enable operation	Set zero shift	Manufac-turer-specific	Parameter No.				
DEVICE CONTROL COMMAND	15	14	13	12	11	10	9	
ENABLE OPERATION	0>1	0	x	0				
PARAMETERIZATION	0	0	x	1 . . . 15				Profile Class 3

Parameter No.

In class 3 encoders, bits 9 to 12 indicate the number of the parameter that is transmitted via bits 0 to 24 of the process data channel.

Comment: the test system for conformance testing expects an acknowledgment within 1 second.

(see Chapter: Parameter Transmission for Profile 3 Encoders).

Set Zero Shift

If the bit = 1, the 'zero shift' parameter is set to the value ('preset value' - 'API'). If the new position actual value in the next bus cycle is not yet available, the encoder must switch-over to the PARAMETERIZATION state.

Manufacturer-Specific

Bits 0 - 8 are reserved in class 3 encoders. In class 4 encoders, these bits are manufacturer-specific.

Bit 13 is manufacturer-specific in class 3 and class 4 encoders.

Access

- class 3 : read only
- class 4: read and write

Object class:	Mandatory
Access:	Read and write
Process data mapping:	PE and PA data
Unit:	None
Value range:	Octet string
Mandatory range:	-
Default value:	

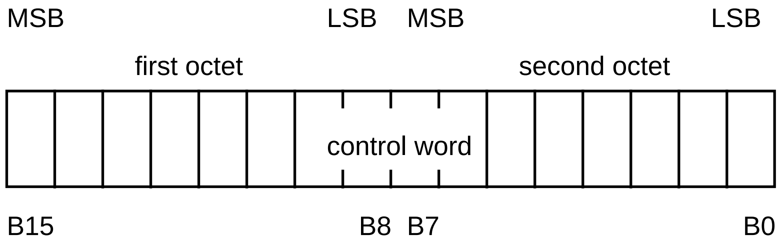
Mapping of the Device Function onto Communication

Object description: ’control word’ (see Table 1)

Table 1: Object description: ’control word’

Object Attribute	Value hex	Meaning
Index	6040	Control word
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	02	2 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Mapping of the control word to the octet string:



5.1.3. Status Word

Information concerning the state of the device and messages are shown in the status word.

Status Word for Profile Class 3

Bit	Name	Mandatory	Bit No. in PD Channel
0	Reserved		
1	Reserved		
2	Reserved		
3	Reserved		
4	Reserved		
5	Reserved		
6	Reserved		
7	Reserved		
8	Reserved		
9	Parameter no. or malfunction code	X	25
10	Parameter no. or malfunction code	X	26
11	Parameter no. or malfunction code	X	27
12	Parameter no. or malfunction code	X	28
13	Manufacturer-specific		29
14	Parameterization	X	30
15	Invalid position actual value	X	31

Status Word for Profile Class 4

Bit	Name	Mandatory
0	Manufacturer-specific	
1	Manufacturer-specific	
2	Manufacturer-specific	
3	Manufacturer-specific	
4	Manufacturer-specific	
5	Manufacturer-specific	
6	Manufacturer-specific	
7	Manufacturer-specific	
8	Manufacturer-specific	
9	Reserved	X
10	Reserved	X
11	Reserved	X
12	Reserved	X
13	Manufacturer-specific	
14	Parameterization	X
15	Invalid position actual value	X

Device States

The device states are shown in the status word by the following bit combinations:

	Control word (bit)						
	Invalid position actual value	Parameter -ization	Manufac- turer specific	Parameter No.			
STATE	15	14	13	12	11	10	9
OPERATION	0	0	x	0			
PARAMETERIZATION	1	1	X	1 . . . 15			
MALFUNCTION	1	0	x	1 . . . 15			

Parameter No or Malfunction Code

In class 3 encoders the number of the parameter that was transmitted to the encoder is acknowledged via bits 9 to 12, or a malfunction code is transmitted (in "malfunction" state).

(see Chapter, Parameter Transmission for Profile 3 Encoders)

Manufacturer-Specific

Bits 0 and 8 to 13 are manufacturer specific.

Object class:	Mandatory
Access:	Read only
Process data mapping:	PE data
Unit:	None
Value range:	Octet string
Mandatory range:	-
Default value:	-

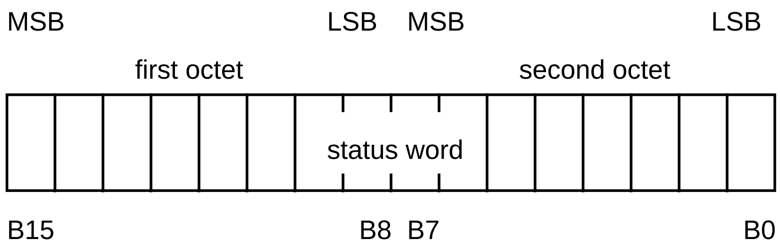
Mapping of Device Function onto Communication

Object description: ’status word’ (see Table 2)

Table 2: Object description: ’status word’

Object Attribute	Value hex	Meaning
Index	6040	Status word
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	02	2 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Mapping of the status word onto the octet string:



5.1.4. Parameter Transmission for Profile Class 3 Encoders

The parameters of a profile class 3 encoder can be transmitted to the encoder via bits 0 to 24 of the process data channel. To do this, the encoder must be set to the parameterization state. This is achieved by outputting a parameter number unequal to zero on bits 9 to 12 of the control word (bits 25 to 28 of the process data channel).

Parameter No. B12 B9	Function	Mandatory
0 0 0 0	Output in "operation" state	x
0 0 0 1	Resolution: sub-parameter steps	x
0 0 1 0	Resolution: sub-parameter revolutions	x
0 0 1 1	Position actual value coding	x
0 1 0 0	Preset value	x
0 1 0 1	Zero shift	x
0 1 1 0	Offset	
0 1 1 1	Reset encoder	

When the ’reset encoder’ parameter is transmitted to the encoder, all parameters are reset to their default values.

The encoder switches over to the "parameterization" state and indicates the successful transmission of the parameter with the return of the corresponding parameter number. The new parameter takes effect after the user has set the encoder to the ’operation’ state with the ’enable operation’ device control command.

If it was not possible for the parameter to take effect, the encoder switches over to the ’malfunction’ state - after the user has sent the ’enable operation’ device control command - and outputs a malfunction number on bits 9 to 12 of the status word (bit 25 to 28 of the process data channel).

Malfunction No. b12 b9	Meaning
0 0 0 0	No malfunction
0 0 0 1	Invalid parameters from the host
0 0 1 0	Unknown parameter number
0 0 1 1	Parameter loss
1 1 0 0	Manufacturer-specific malfunction code
1 1 0 1	Manufacturer-specific malfunction code
1 1 1 0	Manufacturer-specific malfunction code
1 1 1 1	Manufacturer-specific malfunction code

Comment:
If the devices execute a parameter check on starting up (e.g., EEPROM check) and parameter sets with errors are recognized,the "parameter loss" error code is indicated.

EXAMPLE 1: Example of a Parameter Transmission

	Host to Encoder			Encoder to Host			Comment
	Control Word			Status Word			
	D31	D25-D28	D0-D24	D30-D31	D25-D28	D0 - D24	
1	0	0	x	0 0	0	Actual value	Normal operation
2	0	P. No.	Parameter	0 0	0	Actual value	Host transmits parameter to encoder, Encoder does not yet react
3	0	P. No.	Parameter	0 0	0	Actual value	Host continues to wait for acknowledgment from encoder
4	0	P. No.	Parameter	1 1	1)	x	Encoder has accepted the parameter and begins processing
5	0	P. No.	Parameter	1 1	1)	x	Parameter processing still running in encoder
6	0	P. No.	Parameter	1 1	P. No.	x	Processing of parameter is completed. Encoder remains in "parameterization" state
7	1	0	0	1 1	P. No.	x	Device control command "enable operation" from host to the encoder. Encoder does not yet react
8	1	0	0	0 0	0	Actual value	Encoder once more in "operation" state
9	0	0	0	0 0	0	Actual value	Normal operating mode once more reached by both devices

1): You must make sure that when a parameter is transmitted, the same parameter number as that which has already been acknowledged by the encoder is not sent. The repeated transmission of the same parameter is not valid without first leaving the "parameterization" state.

The parameter number must be consistent.

To transmit several parameters, repeat steps 4 to 6.

When transmitting a parameter, make sure that the parameter number is not output before the parameter.

When invalid or inconsistent parameters are sent, the decoder goes into the "malfunction" state when trying to enable operation.

EXAMPLE 2: Sequence when Transmitting Inconsistent Parameters

	Host to Encoder			Encoder to Host			Comment
	Control Word			Status Word			
	D31	D25-D28	D0-D24	D30-D31	D25-D28	D0 - D24	
1	0	0	x	0 0	1)	Actual value	Normal operation
2	0	1	20 000	0 0	1)	Actual value	Host sends the first value for programming of the resolution (number of steps)
3	0	1	20 000	0 0	1)	Actual value	Host continues to wait for acknowledgment from encoder
4	0	1	20 000	1 1	1)	x	Encoder has accepted the parameter and begins processing
5	0	1	20 000	1 1	1)	x	Parameter processing still running in encoder
6	0	1	20 000	1 1	1	x	Processing of parameter is completed. Encoder remains in "parameterization" state
7	0	2	1	1 1	1	x	Host sends the second value for programming of the resolution (number of revolutions)
8	0	2	1	1 1	1	x	Host continues to wait for acknowledgment from encoder
9	0	2	1	1 1	2	x	Processing of parameter is completed. Encoder remains in "parameterization" state
10	1	0	0	1 1	2	x	Device control command "enable operation" from host to the encoder. Encoder does not yet react
11	1	0	0	0 1	1	x	Encoder switches to "malfunction" state, the malfunction code is "1". The position actual value is output on the process data channel.

An attempt has been made here to set the resolution of the encoder to 20,000 steps / revolution. The encoder reacts to this invalid parameter combination by switching over to the "malfunction" state and outputs the malfunction code "1".

5.2. Encoder Function

The encoder function (see Figure 7) comprises functions in which the position detection of the encoder is described.

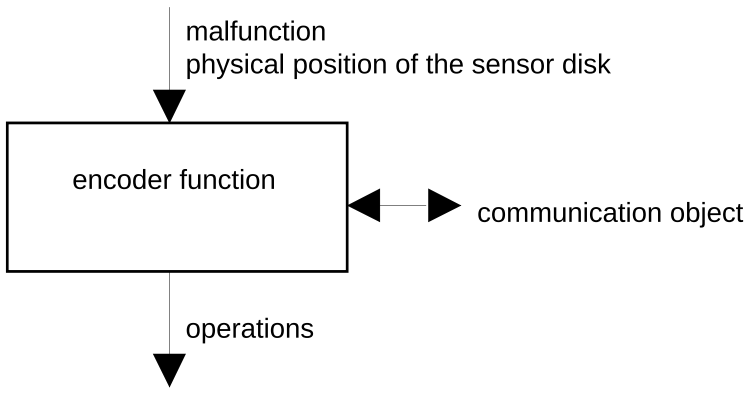


Figure 7: Encoder function

The encoder function consists of the following part functions:

- step function;
- scaling function;
- limiting function;
- actual value correction function.

The encoder functions can be parameterized with the following parameters:

- resolution (steps / revolution);
- position actual value, min/max;
- offset ;
- default value.

The encoder functions provide the following output parameters:

- position actual value;
- absolute position actual value ;
- zero shift;
- measuring step;
- total step number.

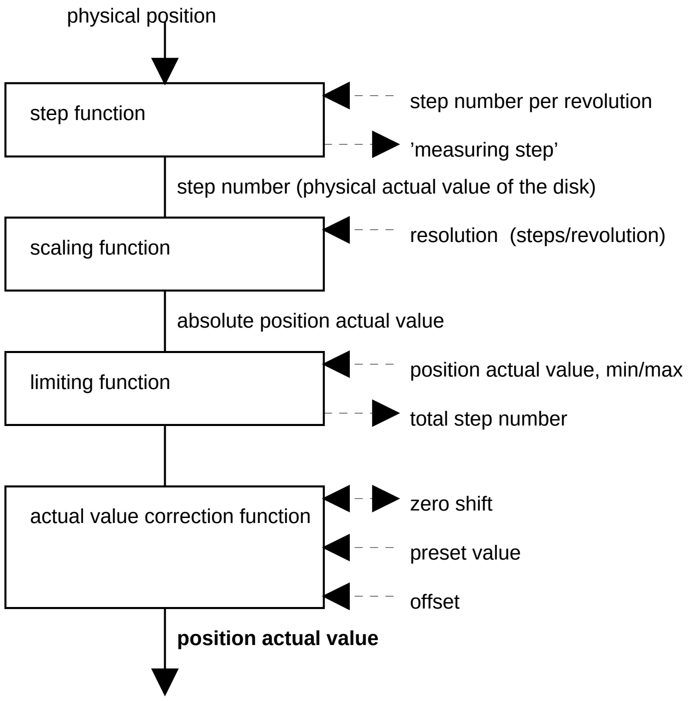


Abb. 8: Part Functions of the Encoder Function

The position actual value is calculated by the formula:

$$PA = \frac{SN}{SNR} * \frac{S}{R} + ZS + O$$

where:

PA position actual value;

SN step number;

SNR step number per revolution;

S Step number;

R number of revolutions;

ZS zero shift;

O offset.

Profile Class Parameter	1	2	3	4
Step number per revolution	Dependent on the encoder disk			
Resolution	Step number per revolution		Parameterizable	
Zero shift	0	0	Parameterizable	
Default value	0	0	Parameterizable	
Offset	0	0	Parameterizable	

5.2.1. Step Function

The step function (see Figure 9) is realized by the encoder disk.

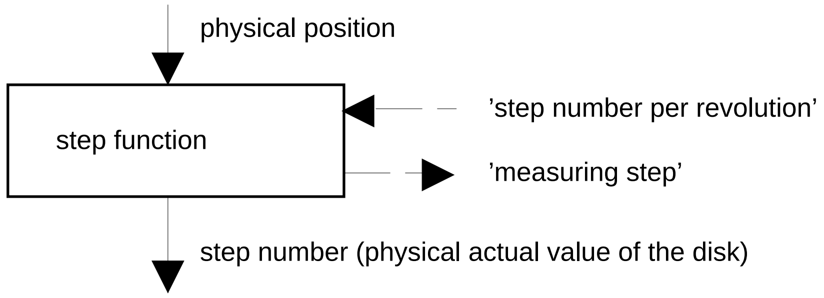


Figure 9: Step function

The step number is calculated by the formula:

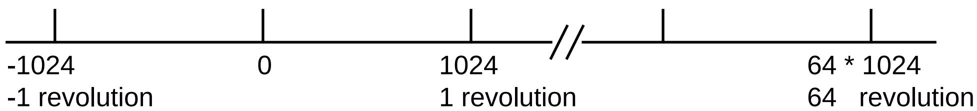
$$SN = R * SNR$$

where:

SN	step number;
R	number of revolutions.
SNR	step number per revolution.

'Step Number per Revolution'

The 'step number per revolution' indicates after how many measuring steps the encoder disk has carried out one revolution:



Object class:	Mandatory
Access:	Read only
Process data mapping:	Not possible
Unit:	Steps
Value range:	Unsigned 16
Mandatory range:	Unsigned 16
Default value:	-

’Step Number

The step number is the physical position of the encoder related to its absolute zero point of the encoder disk.

Object class:	Optional
Access:	Read only
Process data mapping:	PE data
Unit:	Steps
Value range:	Octet string
Mandatory range:	Octet string
Substitute value:	0

Measuring Step

The measuring step is the smallest measurable angle. The unit is degrees or rad. This parameter can only be read out

Object class:	Optional
Access:	Read only
Process data mapping	Not possible
Unit	Degrees or rad (see parameter description data)
Value range:	Unsigned32
Mandatory range:	Unsigned32
Substitute value:	-

Mapping of Device Function onto Communication

Object description: ’step number per revolution’ (see Table **Fehler! Textmarke nicht definiert.**)

Table 3: Object description: ’step number per revolution’

Object Attribute	Value hex	Meaning
Index	604D	Step number per revolution
Variable name	-	Not present
Object code	07	Simple variable
Data type index	06	Unsigned 16
Length	02	2 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description ’step number’ (see Table **Fehler! Textmarke nicht definiert.**)

Table 4: Object description ’step number’

Object Attribute	Value hex	Meaning
Index	6045	Step number
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description ’measuring step’ (see Table 5)

Table 5: Object description ’measuring step’

Object Attribute	Value hex	Meaning
Index	6052	’Measuring step’
Variable name	-	Not present
Object code	07	Simple variable
Data type index		
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

5.2.2. Scaling Function

With the scaling function (see Figure 10) the internal numerical value is converted by the encoder disk (step number) into the absolute position actual value. The absolute position actual value is indicated in steps. To do this, the number of steps from the encoder disk is multiplied by the 'resolution' parameter and divided by the number of 'steps per revolution'. The step size is parameterized with the resolution.

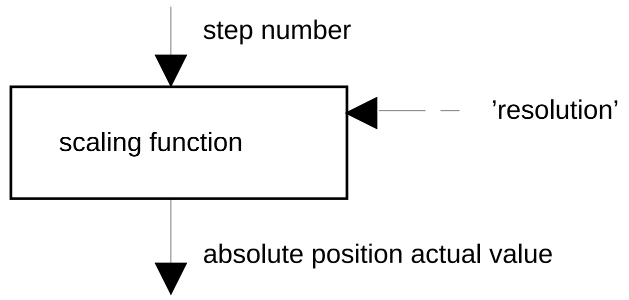


Figure 10: Scaling Function

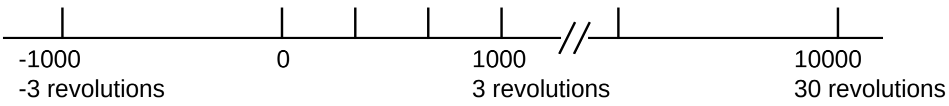
The absolute position actual value is calculated by the formula:

$$APA = \frac{SN}{SNR} * \frac{S}{R}$$

where:

APA	the absolute position actual value;
SN	the step number;
SNR	the step number per revolution;
S	the number of steps;
R	the number of revolutions.

The resolution indicates after how many steps how many revolutions have been carried out by the encoder disk:



’Resolution’

The ’resolution’ parameter is used in the function to change the step size. The resolution results from the desired number of steps that the encoder is to output at a desired number of revolutions. The resolution is composed of two sub-parameters and is calculated by the formula:

$$RES = \frac{S}{R}$$

where:

RES	the resolution;
S	the desired number of steps;
R	the desired number of revolutions.

Steps:

The sub-parameter ’steps’ indicates the desired number of steps.

Revolution:

The sub-parameter ’revolution’ indicates the number of revolutions.

Object class:	Mandatory
Access:	Read and write (write only in class 3)
Process data mapping	Not possible
Unit	1
Value range:	1 to unsigned32 (for all sub-parameters)
Mandatory range:	No (for all sub-parameters)
Default value:	Number of measuring steps per revolution / 1 or value securely stored in case of voltage breakdown

For class 3, the sub-parameters "steps" and "revolutions" are mapped on the bits 0 to 24 of the process data channel.

’Absolute Position Actual Value’

Indicates the number of steps from the encoder zero point.

Object class:	Optional
Access:	Read only
Process data mapping:	PE data
Unit:	1
Value range:	Octet string
Mandatory range:	No
Substitute value:	-

Mapping of Device Function onto Communication

Object description: ’resolution’ (see Table 6)

Table 6: Object description: ’resolution’

Object Attribute	Value hex	Meaning
Index	604B	’Resolution’
Variable name	-	Not present
Object code	08	Array
Number of elements	02	2 elements
Data type index	06	Unsigned 16
Length	02	2 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description ’absolute position actual value’ (see Table 7)

Table 7: Object description ’absolute position actual value’

Object Attribute	Value hex	Meaning
Index	6046	Absolute position actual value
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

5.2.3. Limiting Function

The value range of the encoder's position value is limited to a minimal and a maximum value by the limiting function (see Figure 11). If the position value increases above the maximum value by rotating the encoder beyond the maximum value, the device contines with the minimum value.

COMMENT: This function is not supported by class 1 to class 3 encoders.

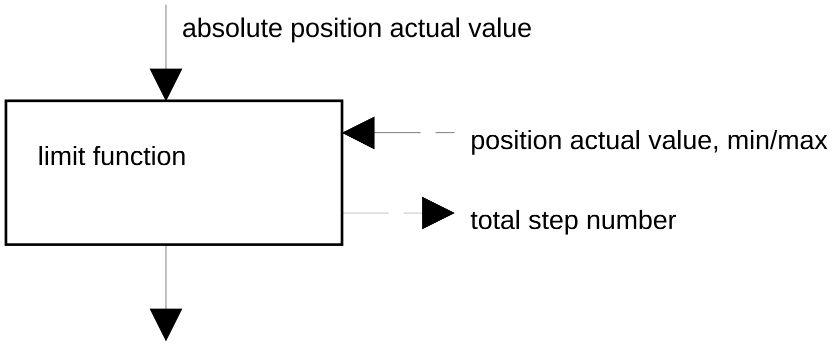


Figure 11: Limiting function

'Position Actual Value, Min/Max'

This parameter indicates the minimal and maximum value range of the position value of the encoder.

Object class:	Optional
Access:	Read and write
Process data mapping	Not possible
Unit	See 'quantity index' and 'unit index'
Value range:	Integer32 (for all sub-parameters)
Mandatory range:	No (for all sub-parameters)
Substitute value:	Value securely stored in case of voltage breakdown

'Total Step Number'

The total step number is the number of all measuring steps that the encoder can output.

Object class:	Optional
Access:	Read only
Process data mapping:	Not possible
Unit:	Steps
Value range:	Unsigned32
Mandatory range:	No
Substitute value:	-

Mapping of Device Function onto Communication

Object description: ’position actual value, min/max’ (see Table 8)

Table 8: Object description ’position actual value, min/max’

Object Attribute	Value hex	Meaning
Index	604C	Position actual value, min/max
Variable name	-	Not present
Object code	08	Array
Number of elements	02	2 elements
Data type index	04	Integer32
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description ’total step number’ (see Table 9)

Table 9: Object description ’total step number’

Object Attribute	Value hex	Meaning
Index	6051	Total step number
Variable name	-	Not present
Object code	07	Simple variable
Data type index	07	Unsigned32
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

5.2.4. Actual Value Correction Function

The actual value correction function (see Figure 12) sets the position actual value to a desired value. With the zero shift parameter, the absolute position actual value is shifted by a desired value.

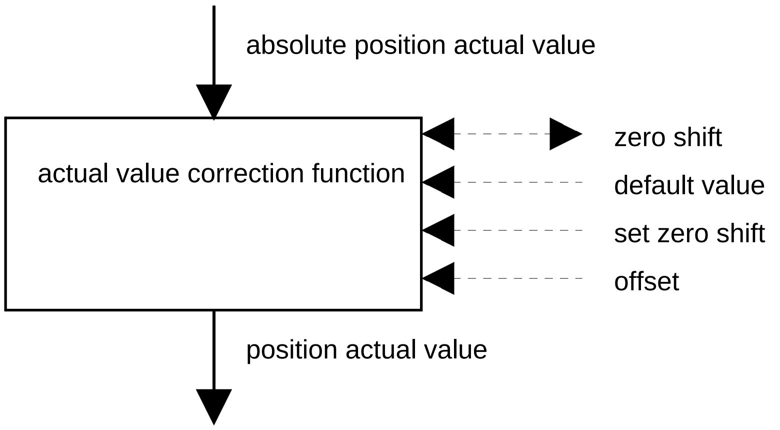


Figure 12: Actual value correction function

The position actual value is calculated by the formula:

$$PA = APA + ZS + O$$

Where:

PA	the position actual value;
APA	the absolute position actual value;
ZS	the zero shift;
O	the offset.

’Position Actual Value’

The position actual value contains the current position.

Object class:	Mandatory
Access:	Read only
Process data mapping	PE data
Unit	See ’quantity index’ and ’unit index’
Value range:	Octet string
Mandatory range:	Octet string
Substitute value:	-

’Position Actual Value 25’

The position actual value 25 is the parameter that is mapped onto the process input data in class 3 devices. It contains the actual position in bit 0 to bit 24. Status bits 9 to 15 are contained in bit 25 to bit 31.

b31 b25	b24 b0
Status word b9-15	Position actual value b0-24

Object class:	Mandatory
Access:	Read only
Process data mapping	PE data
Unit	See ’quantity index’ and ’unit index’
Value range:	Octet string
Mandatory range:	Octet string
Default value:	-

’Positions Actual Value 28’

The position actual value 28 is the parameter that is mapped onto the process input data in class 4 devices. It contains the actual position in bit 0 to bit 27. Status bits 12 to 15 are contained in bit 28 to bit 31.

b31 b28	b27 b0
Status word b12-15	Position actual value b0-27

Object class:	Mandatory
Access:	Read only
Process data mapping	PE data
Unit	See ’quantity index’ and ’unit index’
Value range:	Octet string
Mandatory range:	Octet string
Default value:	-

’Preset Value’

The value of the ’preset value’ parameter is used to determine the zero shift.

Object class:	Mandatory
Access:	Read and write (write only in class 3)
Process data mapping	Not possible
Unit	See ’quantity index’ and ’unit index’
Value range:	Integer32
Mandatory range:	Integer32
Default value:	Value securely stored in case of voltage breakdown

Set Zero Shift

The control command that is triggered by the ’set zero shift’ control bit sets the ’zero shift’ parameter to the value that is calculated by the following formula:

$$ZS = PV - APA$$

Where:

ZS the zero shift ;
PV the preset value;
APA the absolute position actual value.

’Zero Shift’

The zero shift parameter contains the difference of the system zero point to the zero point of the encoder.

Object class:	Mandatory
Access:	Read and write (write only in class 3)
Process data mapping	Not possible
Unit	See ’quantity index’ and ’unit index’
Value range:	Integer32
Mandatory range:	Integer32
Default value:	Value securely stored in case of voltage breakdown

'Offset'

The 'Offset' parameter is an additional possibility for shifting the position actual value.

Object class:	Mandatory
Access:	Read and write (write only in class 3)
Process data mapping	Not possible
Unit	See 'quantity index' and 'unit index'
Value range:	Integer32
Mandatory range:	Integer32
Default value:	Value securely stored in case of voltage breakdown

Mapping of Device Function onto Communication

Object description: 'position actual value' (see Table 10)

Table 10: Object description: 'position actual value'

Object Attribute	Value hex	Meaning
Index	6042	Position actual value
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’position actual value 25’ (see Table 11)

Table 11: Object description: ’position actual value 25’

Object Attribute	Value hex	Meaning
Index	6043	Position actual value 25
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’position actual value 28’ (see Table 12)

Table 12: Object description: ’position actual value 28’

Object Attribute	Value hex	Meaning
Index	6043	Position actual value 28
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’preset value’ (see Table 13)

Table 13: Object description: ’preset value’

Object Attribute	Value hex	Meaning
Index	6044	Preset value
Variable name	-	Not present
Object code	07	Simple variable
Data type index	04	Integer32
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’zero shift’ (see Table 14)

Table 14: Object description: ’zero shift’

Object Attribute	Value hex	Meaning
Index	604E	Zero shift
Variable name	-	Not present
Object code	07	Simple variable
Data type index	04	Integer32
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: 'offset' (see Table 15)

Table 15: Object description: 'offset'

Object Attribute	Value hex	Meaning
Index	604E	Offset
Variable name	-	Not present
Object code	07	Simple variable
Data type index	04	Integer32
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0003	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

5.2.5. Malfunction Function

The malfunction function (see Figure 13) administers the ’malfunction code’ parameter. The ’malfunction code’ parameter is set to the corresponding value in case of a device malfunction (see malfunction list). By the enable operation action of the device control, the parameter is set to the value 0.

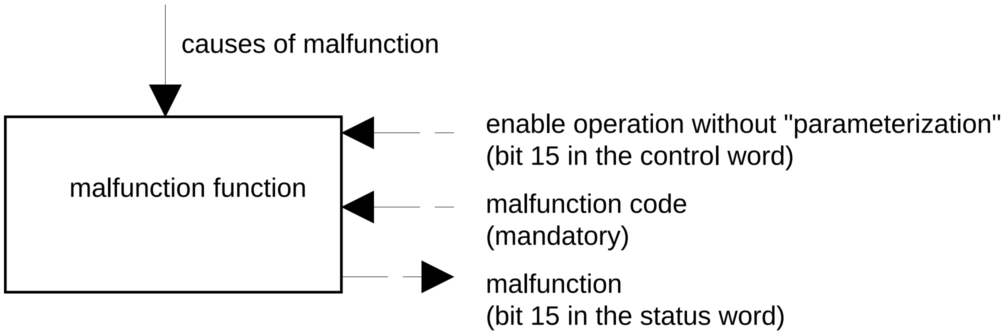


Figure 13: Malfunction function

’Malfunction Code’

The ’malfunction code’ is displayed as an octet string with a length of 2 bytes. The coding is hierarchical and ranges from a coarse to an increasingly finer differentiation.

Bit	Grouping
15 . . . 12	Main groups
11 . . . 8	Sub-groups
7 . . . 0	Details

When the encoder is in the malfunction state the parameter contains a value unequal to 0. When the encoder is not in the malfunction state, the parameter contains the value 0.

If there is exactly one cause of malfunction the value assigned to this cause can be read out unchanged in the ’malfunction code’ parameter until the malfunction state is left. The malfunction state is left when the cause of the malfunction is eliminated and the enable operation command is given.

If there are several causes of malfunction simultaneously, one of them is displayed in the ’malfunction code’ parameter. If only the cause of malfunction that is displayed is eliminated and the enable operation command is given, the malfunction state is not left, due to the remaining causes of malfunction. One of these causes of malfunction is then displayed in the ’malfunction code’ object.

Object class:	Mandatory
Access:	Read only
Process data mapping:	Not possible
Unit:	None
Value range:	Octet string
Mandatory range:	-
Default value:	0

Table 16: Malfunction codes and causes of malfunctions

Code hex	Meaning
0000	**No disturbance**
1000	**General disturbance**
2000	**Current**
3000	**Voltage**
3100	**Mains voltage**
3110	surge voltage
3120	undervoltage
3130	phase failure
3140	mains frequency
3200	**Internal device voltage**
3210	surge voltage
3220	undervoltage
3230	charging error
3300	**Output Voltage**
3310	output surge voltage
4000	**Temperature**
5000	**Device Hardware**
5100	**Supply**
5110	low voltage supply
5111	supply +/- 15 V
5112	supply + 24 V
5113	supply + 5 V
5200	**Control**
5300	**Operating Unit**
6000	**Device Software**
6010	software reset (watchdog)
6100	**Internal Software**
6300	**Data Set**
6310	parameter loss
6320	parameter error
7000	**Additional Assemblies**
8000	**Monitoring**
9000	**External Malfunction**
F000	**Additional Functions**

Mapping of Device Function onto Communication

Object description: ’malfunction code’ (see Table 17)

Table 17: Object description: ’malfunction code’

Object Attribute	Value hex	Meaning
Index	603F	Malfunction code
Variable name	-	Not present
Object code	07	Simple variable
Data type index	0A	Octet string
Length	02	2 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

5.3. Sensor/Actuator Function

5.3.1. Communication Functions

See Sensor/Actuator Profile 12, Chap. ’Communication Functions’.

COMMENT: The initiate service is the request of a connection establishment. The ’Profile Number’ service parameter must be set to the value 0071 hex by the initiator of the connection establishment.

5.3.2. Device Information

See Sensor/Actuator Profile 12, Chap. ’Device Information’ .

6. Data Structures

The data structures of all user data are gathered in this chapter.

The parameters of an encoder are stored in an object dictionary. This object dictionary is used to describe the parameters. It contains details of the index, data type, object type, access rights, etc. The index is used to address the parameter when read and write operations. This object dictionary can be read out with the ’Read Communication Object List’ function.

Composition of the Object Dictionary (see Table 18)

Table 18: Composition of the object dictionary

Index	Object Dictionary
0000	OD object description
0001 001F	(PROFIBUS DIN 19245/Part 2) Static type index
0020 003F	(Profile) Static type index
0040 005F	(Reserved for manufacturer) Static type index
. . .	. . .
2000 5FFF	(Reserved for manufacturer) Static object index
6000 603F	(Server profile corresponding to the device) Static object index
6040 9FFF	(Profile) Static object index
A000 BFFF	(Reserved for manufacturer) Dynamic variable list index
C000 DFFF	(Profile) Dynamic variable list index
E000 EFFF	(Reserved for manufacturer) Dynamic program invocation index
F000 FFFF	(Profile) Dynamic program invocation index

Object Description of the Zero Objects

This object description is stored in the indices for which there is no object (e.g., non-supported optional objects).

Index	xxxx hex	=
Object code	0 hex	= Zero object

A list of all parameters addressable by the communication is contained in Table 19.

Table 19: List of all parameters addressable by the communication

Index	Type	Object	Name	m/o
6000	PDB structure	record	PE data description	o
6001	PDB structure	record	PE data description	o
6002	Boolean	var	PE data enable	o
6003	Unsigned16	var	PD monitoring time	o
6004	Integer16	var	PD monitoring selection code	o
6005	Unsigned16	var	C monitoring time	o
6006	Integer16	var	C monitoring selection code	o
6007	Integer16	var	Connection abort selection code	o
6008	Unsigned16	var	ID number	o
6009	Unsigned8	var	Parameter set identifier	o
600A	Unsigned32	var	Serial number	o
600B	Date	var	Date of calibration	o
600C	Visible str.	var	Device description	o
600D	Visible str.	var	Model description	o
600E	Date	var	Date of parameterization	o
...				
603E		zero		m
603F	Octet string	var	Malfunction code	m
6040	Octet string	var	Control word	m
6041	Octet string	var	Status word	m
6042	Octet string	var	Position actual value	m
6043	Octet string	var	Position actual value 25	m
6044	Octet string	var	Position actual value 28	m
6045	Integer32	var	Step number	o
6046	Integer32	var	Absolute position actual value	o
6047	Integer8	var	Position actual value coding	m
6048	Unsigned8	var	Quantity index	o
6049	Integer8	var	Unit index	o
604A	Integer32	var	Total offset	o
604B	Unsigned32	array	Resolution	m
604C	Integer32	array	Position actual value, min/max	o
604D	Unsigned16	var	Step number per revolution	o
604E	Integer32	var	Zero shift	m
604F	Integer32	var	Preset value	m
6050	Integer32	var	Offset	o
6051	Unsigned32	var	Total step number	o
6052	Unsigned32	var	Measuring step	o

m = mandatory o = optional

6.1. Encoder Functions

The structure of application data of the communication device is described in this chapter:

- position value description data ;
- parameter description data of the device parameter ;
- limit values of the parameter description data;
- default setting of the parameter description data.

6.1.1. Position Value Description Data

The display of the position values that an encoder provides is determined by the position value description data.

The position value description data is stored in the following parameters:

- position actual value coding;
- quantity index;
- unit index;
- total offset (measuring range shift);
- resolution counter/denominator;
- position actual value, min/max.

These parameters have fixed preset values in non-parameterizable encoders and must be indicated in the data sheet. In parameterizable encoders, display of the position values can be set by parameterizing the position value description data.

’Position Actual Value Coding’

This parameter defines how the parameter, ’step number’, ’absolute position actual value’ and ’position actual value’ of the encoder is coded and describes the code sequence.

	Meaning
-128 to -1	Manufacturer-specific
0	Reserved
1	Signed dual code (integer) Ascending code in ’right’ direction of rotation (clockwise when looking at the shaft
2	Signed dual code (integer) Ascending code in ’left’ direction of rotation (anticlockwise when looking at the shaft)
3	Unsigned dual code (unsigned) Ascending code in ’right’ direction of rotation (clockwise when looking at the shaft)
4	Unsigned dual code (unsigned) Ascending code in ’left’ direction of rotation (anticlockwise when looking at the shaft)
5	Unsigned gray code Ascending code in ’right’ direction of rotation (clockwise when looking at the shaft)
6	Unsigned gray code Ascending code in ’left’ direction of rotation (anticlockwise when looking at the shaft)

Display of Signed Dual Codes

PD bit 24 Sign	PD bit 23 . . . PD bit 0 Dual code

There is a division into a positive and a negative range:

Positive range 0 ... 2^{24-1}	Negative range -2^{24-1}... -1

The size of the respective range depends on the number of steps. In case of an even or odd number of steps, the sizes of the two ranges are different.

Calculation of the ranges for an even step number:

Positive range: 0 to (steps / 2) - 1 Negative range: -(steps / 2) to -1

Example: even step number

Steps = 12
Positive range = 0 to (12/2)-1 => 0 to 5
Negative range = -(12/2) to -1 => -6 to -1
The following output results: 0 1 2 3 4 5 / -6 -5 -4 -3 -2 -1

Calculation of the ranges for an uneven number of steps:

Positive range: 0 to (steps -1) / 2 Negative range: (1 - steps) / 2 to -1

Example: odd step number

Steps = 9
Positive range = 0 to (9/2)-1 => 0 to 4
Negative range = -(9/2) to -1 => -4 to -1
The following outputresults: 0 1 2 3 4 / -4 -3 -2 -1

Comment:
In coding without a sign, a rotation is made in the opposite direction, starting with the maximum displayable value and then continually decreasing in value.

Object class:	Mandatory
Access:	Read and write
Process data mapping	Not possible
Unit	None
Value range:	Integer8
Mandatory range:	Profile-specific code
Default value:	Value securely stored in case of voltage breakdown

’Quantity index’’

The ’quantity index’ indicates the physical value of the parameters ’step number’, ’absolute position actual value’ and ’position actual value’ in coded form. The value range is 0 to 256. The coding can be found in the table.

Object class:	Optional
Access:	Read and write
Process data mapping	Not possible
Unit	None
Value range:	Unsigned8
Mandatory range:	Profile-specific code
Default value:	Value securely stored in case of voltage breakdown

’Unit Index’

The ’unit index’ indicates the measuring unit of the parameter ’step number’, ’absolute position actual value’ and ’position actual value’ in coded form. The unit index has the value range -127 to 128. The coding can be found in the table.

Object class:	Optional
Access:	Read and write
Process data mapping	Not possible
Unit	None
Value range:	Unsigned8
Mandatory range:	Profile-specific code
Default value:	Value securely stored in case of voltage breakdown

’Total Offset’

This parameter indicates the total offset. This is comprised of the zero shift and the offset.

Object class:	Optional
Access:	Read only
Process data mapping	Not possible
Unit	None
Value range:	Integer32
Mandatory range:	Integer32
Default value:	Value securely stored in case of voltage breakdown

’Resolution’

This parameter indicates the number of steps that the encoder outputs on a selected number or revolutions (see scaling function).

’Position Actual Value, Min/Max’

This parameter is comprised of two sub-parameters together. The first sub-parameter indicates the smallest position actual value that the encoder outputs. The second sub-parameter indicates the highest position actual value that the encoder outputs (see limiting function).

EXAMPLE 1:

Parameter	Position Value Description Data	Meaning
Position actual value coding	1	Signed dual code (integer) Ascending code in 'right' direction of rotation (clockwise) seen from the shaft
Quantity index	32	Step
Unit index	0	10-0
Total offset		Zero shift + offset
Resolution counter/denominator	1024 / 1	1024 steps/revolution
Position actual value, min/max	0 / 65535	

Table 20: Quantity index and unit index

Physical Value	Quantity index	Unit	Unit Index
Length	1	Meter	0
		Millimeter	-3
		Kilometer	3
		Micrometer	-6
Angle	12	Radiant	0
		Second	75
		Minute	76
		Degree	77
		Centesimal degree	78
Steps	32	Steps	0
Encoder resolution	33	Steps/Revolution	0

The value of the unit index thereby corresponds to the respective decimal power of the standard unit, e.g.

Unit index 0 for 10^0

Unit index 3 for 10^3

Unit index -3 for 10^{-3}

etc.

The unit indices listed for SI-compatible units (unit index < 64) only serve as an example. The unit indices for further SI-compatible prefixes (pico etc.) result analogously. Unit indices that are greater than +64 have a special significance that must be determined from the table. These units are, e.g. day, hour, minute or non SI-compatible units such as Fahrenheit.

Mapping of Device Function onto Communication

Object description: ’position actual value coding’ (see Table 21)

Table 21: Object Description: ’Position actual value coding’

Object Attribute	Value hex	Meaning
Index	6047	Position actual value coding
Variable name	-	Not present
Object code	07	Simple variable
Data type index	02	Integer8
Length	01	1 byte
Password	00	None
Access groups	00	None
Access rights	0001	Read all, write all
local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’quantity index’ (see Table 22)

Table 22: Object description: ’quantity index’

Object Attribute	Value hex	Meaning
Index	6048	Quantity index
Variable name	-	Not present
Object code	07	Simple variable
Data type index	02	Integer8
Length	01	1 byte
Password	00	None
Access groups	00	None
Access rights	0001	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’unit index’ (see Table 23)

Table 23: Object description: ’unit index’

Object Attribute	Value hex	Meaning
Index	6048	Unit index
Variable name	-	Not present
Object code	07	Simple variable
Data type index	02	Integer8
Length	01	1 byte
Password	00	None
Access groups	00	None
Access rights	0001	Read all, write all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

Object description: ’total offset’ (see Table 24)

Table 24: Object description: ’total offset’

Object Attribute	Value hex	Meaning
Index	6048	Total offset
Variable name	-	Not present
Object code	07	Simple variable
Data type index	04	Integer32
Length	04	4 bytes
Password	00	None
Access groups	00	None
Access rights	0001	Read all
Local address	xxxx	Manufacturer-specific
Extension	-	Not present

6.1.1.1. Parameter Description Data of the Device Parameters

The parameter description data corresponds to the preset structure in InterBus-S Sensor/Actuator Profile. The parameters of a device function are described with a parameter description. The parameters and the parameter description are stored as device parameters in the device.

Parameter description of the parameter ’step number’:

module type = ’position actual value coding’

quantity index = ’quantity index’

unit index = ’unit index’

offset = -

resolution = -

min/max values = -

Parameter description of the parameter ’step number per revolution’:

module type = -

quantity index = ’quantity index’

unit index = ’unit index’

offset = -

resolution = -

min/max values = -

Parameter description of the parameter ’absolute position actual value’:

module type = ’position actual value coding’

quantity index = ’quantity index’

unit index = ’unit index’

offset = -

resolution = ’resolution’

min/max values = -

Parameter description of the parameter ’position actual value’:

module type = ’position actual value coding’

quantity index = ’quantity index’

unit index = ’unit index’

offset = ’total offset’

resolution = ’resolution’

min/max values = ’position actual value, min/max’

Parameter description of the parameter ’zero shift’:

module type = -
quantity index = ’quantity index’
unit index = ’unit index’
offset = -
resolution = -
min/max values = -

Parameter description of the parameter ’preset value’:

module type = -
quantity index = ’quantity index’
unit index = ’unit index’
offset = -
resolution = -
min/max values = -

Parameter description of the parameter ’offset’:

module type = -
quantity index = ’quantity index’
unit index = ’unit index’
offset = -
resolution = -
min/max values = -

6.1.2. Limit Values of the Parameter Description Data

The limit values of the parameter description data indicate what value a setting parameter can be set to. This data must be indicated in the data sheet.

Position Actual Value Coding

This list indicates which position actual values codings are supported.

Quantity Index

This list indicates in which unit the test value can be represented.

Unit Index

This list indicates the unit indices with which the test value can be represented.

Step Number per Revolution

This value indicates the maximum step number with which a revolution is resolved.

Maximum Number of Revolutions

This parameter indicates the maximum displayable revolutions of the encoder.

Minimum Parameterizable Lowest Position Actual Value, Min

This parameter indicates the smallest value to be output.

Maximum Parameterizable Highest Position Actual Value, Max

This parameter indicates the highest value to be output.

Measuring Step

The measuring step is the smallest measurable angle. The unit is degrees or rad. This parameter can only be read.

6.1.3. Default Setting of the Parameter Description Data

The default setting is the parameter description data that the encoder is set to on delivery. This data is manufacturer-specific and is adopted when the encoder is reset. These values are stored in the ROM of the encoder and cannot be changed by parameterization; they must be indicated in the data sheet.

- position actual value coding
- quantity index
- unit index
- total offset (measuring range shift)
- resolution (steps/revolutions)
- position actual value, min/max

7. Operating Phases of the Application

The possible operation phases of the device are described in this chapter. The chapter is divided into:

- initialization/abort;
- operation;
- startup phase and configuration phase.

7.1. Initialization/Abort

Initialization

Initialization begins after switching on the power or resetting the encoder.

The following actions are carried out:

- initialization of the communication interface;
- initialization of the process data;
- initialization of the parameters.

Initialization of the Communication Interface

See Sensor/Actuator profile 12

Initialization of the Process Data

- Configuration of the process input data and process output data

  The assignment of the process input data and process output data is carried out in accordance with the stored configuration (for default setting see Function Block: Communication Function).

- The process input and output data registers are pre-assigned the value zero.

Initialization of the Parameters

During startup, the following communication objects must be parameterized with the corresponding stored values or - when not available - substitute values.

Communication Object	Substitute Value	
Process data monitoring time	65535	Disabled
Process data monitoring selection code	0	No reaction
Communication monitoring time	65535	Disabled
Communication monitoring selection code	0	No reaction
Connection abort selection code	0	No reaction
Process input data description	4, 6044, 0	Length = 4 byte ’position actual value 28’
Process output data description	2, 6040, 0	Length = 2 byte ’control word’

COMMENT: Communication objects are only supported by class 4 devices.

Abort

The following operations are carried out:

- Process data is reset

When the communication unit and the encoder unit are decoupled, the process input data is set to zero if the encoder unit fails.

7.2. Operation

The following functions are active in operation phase ’operation’:

- device control;
- encoder functions;
- sensor/actuator functions.

8. Communication Profile

8.1. Layer 1

All definitions concerning layer 1 are set in this chapter.

InterBus-S Coupling

InterBus-S coupling can be carried out via the following interfaces:

- remote bus interface;
- local bus interface.

Remote Bus Interface

- Subminiature D 9 pin (male) to the controller
- Subminiature D 9 pin (female)to the end of the bus
- 2 wire ring
- Diagnostics LEDs
    - remote bus control (RC) green
    - remote bus disable (Rbd) red
    - bus active (BA) green
    - transmit (TR) green (only when PCP is implemented)

Local Bus Interface

- Subminiature D 15 pin
- Interface electrically isolated, 500 V AC
- diagnostics LEDs
    - operating voltage (5 V) green

Connection Method

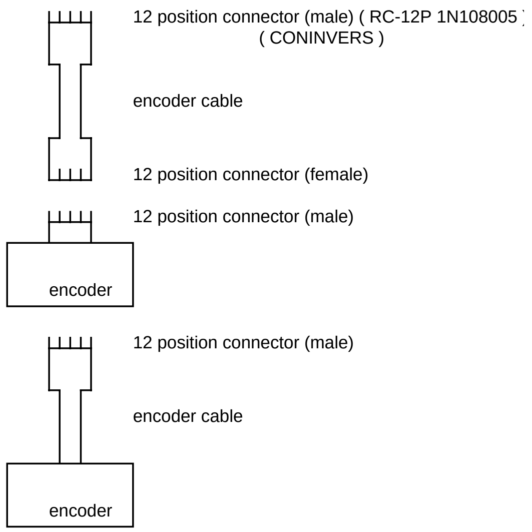


Figure 14: Connection possibilities

Encoder Cable

Plug Pin	Signal	Wire Color
1	DO2	Violet
2	DO2 neg	Black
3	DI2	Red-blue
4	DI2 neg	Gray-pink
5	DO1	Yellow
6	DO1 neg	Green
7	DI1	Gray
8	DI1 neg	Pink
9	RBST neg	White
10	0 V	Blue <
11	GND	Brown | (0 volt of the + UB (24 V))
12	+ UB (24V)	Red <

Connector for Remote Bus

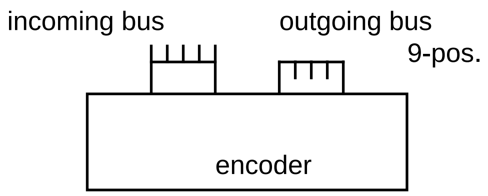


Figure 15: Connector

Connector for passive T-coupler (see Figure 16):

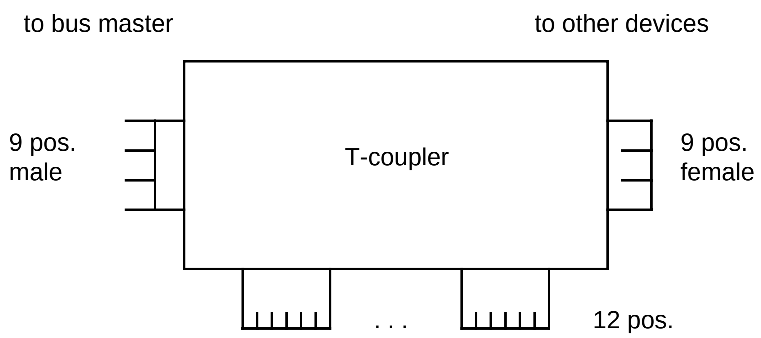


Figure 16: Connector for the passive T-coupler

An incremental transducer cable is prescribed as cable between T-coupler and encoder.

Electrical isolation is realized in the encoder or in the T-coupler.

Possible implementations:

- power unit in the T-coupler;
- supply of a 12V voltage.

8.2. Layer 2

All definitions concerning layer 2 are set in this chapter.

InterBus-S Register

The assignment of the data register of an InterBus-S device, and thereby the addressing on the I/O level, is defined in the following.

Configuration of InterBus-S Registers:

Profile class 1 (see Figure 17):

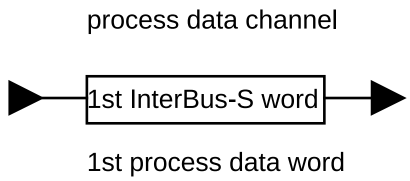


Figure 17: Configuration of InterBus-S registers of profile class 1

Profile classes 2 and 3 (see Figure 18):

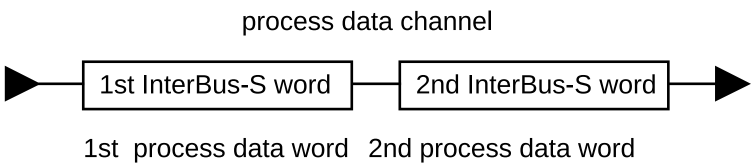


Figure 18: Configuration of InterBus-S registers of profile classes 2 and 3

Profile class 4 (see Figure 19):

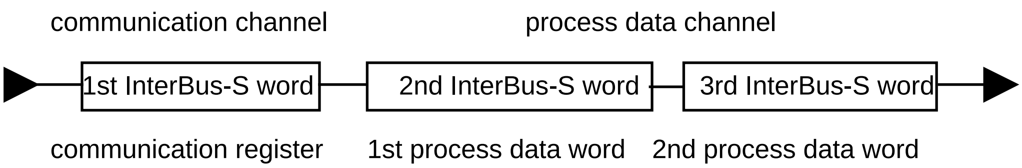


Figure 19: Configuration of InterBus-S registers of profile class 4

Addressing the Process Data (see Figures Fehler! Textmarke nicht definiert.**,** Fehler! Textmarke nicht definiert.**, 21):**

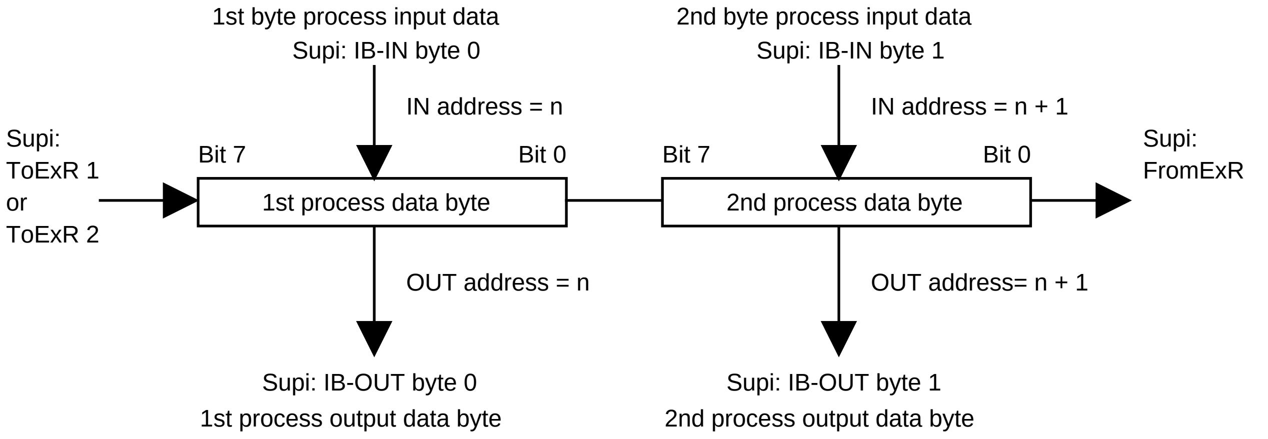


Figure 20: Addressing the process data

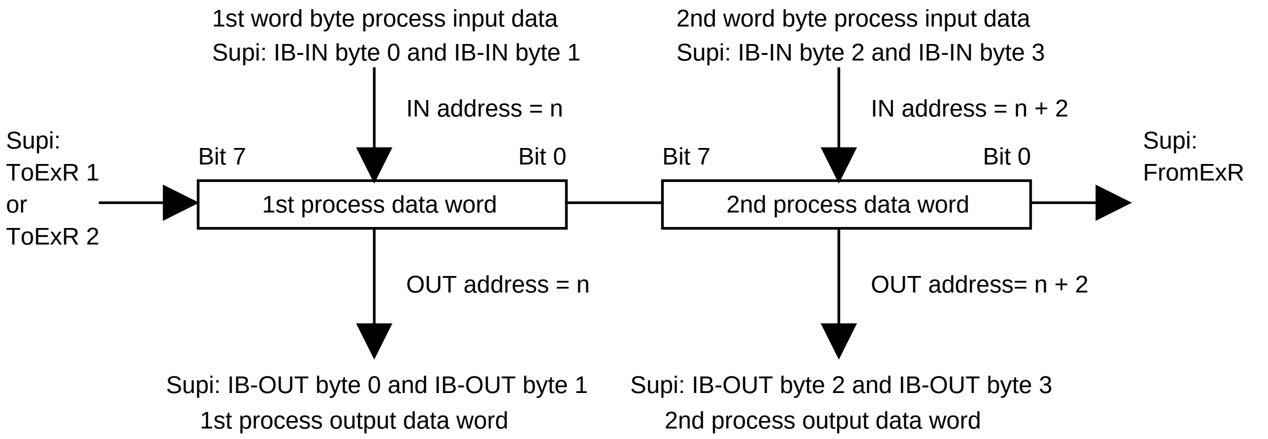


Figure 18: Addressing the process data

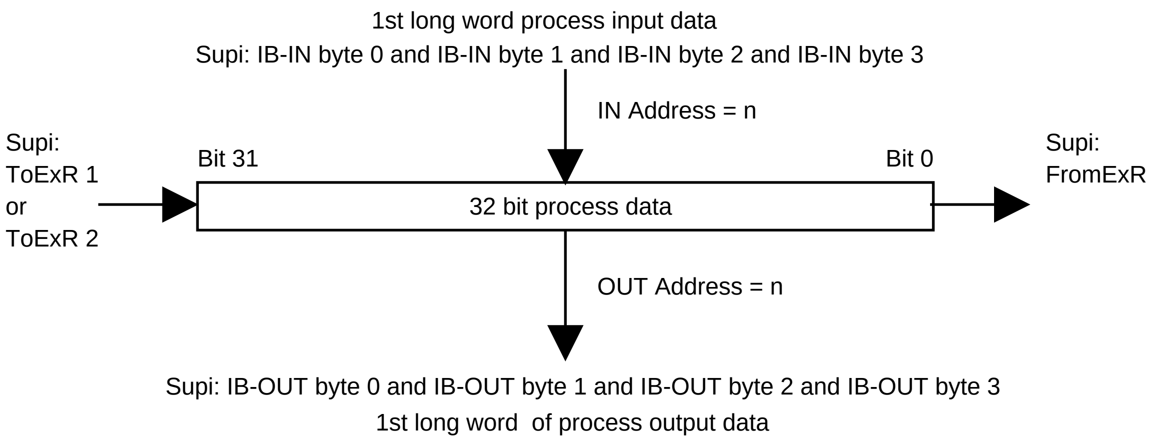


Figure 21: Addressing the process data

Process Data Flow

Process input data is transmitted by the encoder to the bus system.

Process output data is transmitted from the bus system to the encoder.

Identification of InterBus-S Communication Devices (see Table 25)

Table 25: Identification of InterBus-S communication devices

IB-S-Coupling	Class	Number of Words	InterBus-S Device ID-Code		ID-Code
				hex	dec
Remote bus	1	1 IN	0000 0001 0011 0110	0136	54
Remote bus	2	2 IN	0000 0010 0011 0110	0236	54
Remote bus	3	2 IN / 2 OUT	0000 0010 0011 0111	0237	55
Remote bus	4	3 IN / 3 OUT	0000 0011 1111 0111	03F7	247
Local bus	1	1 IN	0000 0001 0110 0110	0166	102
Local bus	2	2 IN	0000 0010 0110 0110	0266	102
Local bus	3	2 IN / 2 OUT	0000 0010 0110 0111	0267	103
Local bus	4	3 IN / 3 OUT	0000 0011 1101 0111	03D7	215

8.3. Layer 7

The parameter channel is used in accordance with the sensor/actuator server communication profile.