5 Operation

5.1 Operation at a glance

Operation at a glance

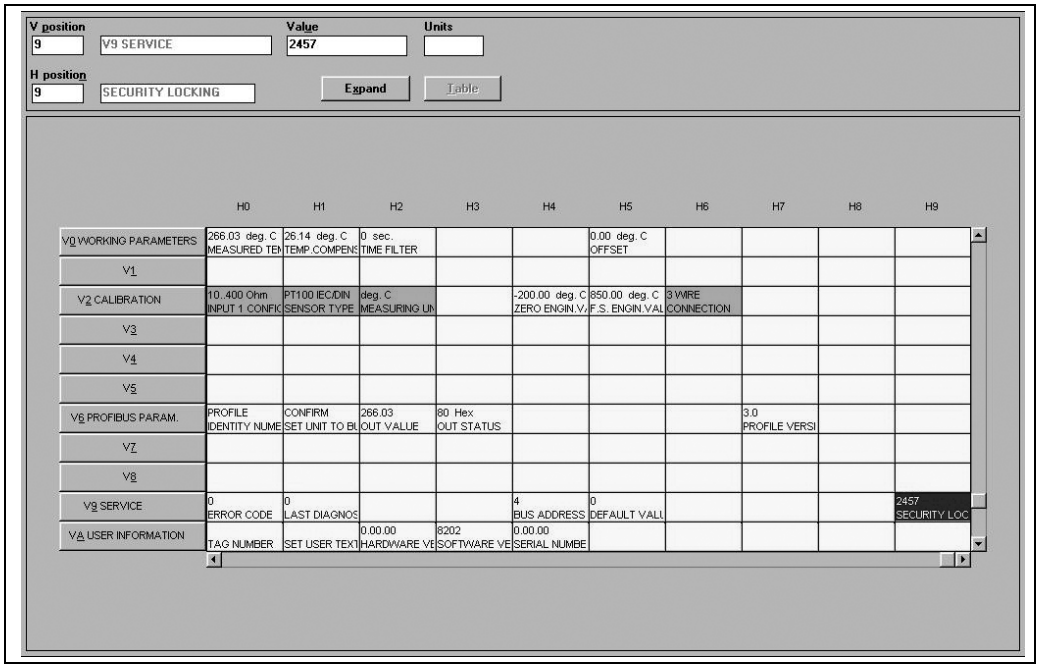

Fig. 5.1: E+H operation programme Commuwin II surface.
Light grey highlighted function fields (V2H0, V2H1, V2H2, V2H6) = Set-up using Quick Setup
Dark grey highlighted function field (V9H1) = Active field

5.2 Communication PROFIBUS-PA®

Communication PROFIBUS-PA®

5.2.1 System architecture PROFIBUS-PA®

PROFIBUS-PA® is an open fieldbus standard to EN 50 170, DIN 19245 and IEC 61158-2, that has been specially designed to the requirements of the process industries.

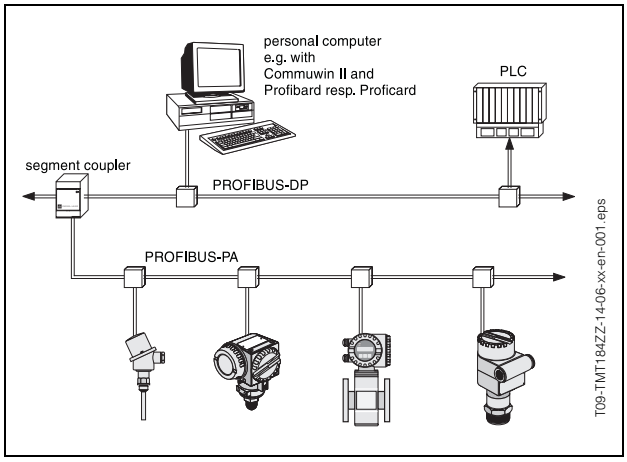

Fig. 5.2: System architecture for PROFIBUS-PA®

In the simplest case the complete measurement system consists of a temperature head transmitter, a segment coupler, a PROFIBUS-PA®- termination resistance, a PLC or a PC with an operation software, e.g. Commuwin II. The maximum number of transmitters per bus segment is dependent on the current consumption of each transmitter, the maximum power of the segment coupler and the required bus length, see BA 198F/00/en.

Normally the following is valid:
- Max 9 temperature head transmitters in EEx ia applications and
- Max 32 temperature head transmitters in non-hazardous areas

can be connected per bus segment.

✎ **Note!**
During project planning please note the current consumption of the temperature head transmitter is 10 ± 1 mA.

5.2.2 Communication partner

In a control system the head transmitter always operates as a slave and, dependent on the application, can exchange data with one or more masters. The master can be a PLC, a DCS or a PC fitted with a PROFIBUS-DP® communication board.

Function blocks
For the description of the function blocks of a unit and definition of unified data access PROFIBUS uses predefined function blocks (→ chap. 5.2.7 to → chap. 5.2.10). The function blocks implemented in fieldbus units transmit information regarding the tasks of a unit within the total automation strategy.

✎ **Note!**
Detailed information regarding the function blocks can be found in the PROFIBUS-PA® Profile for Process Control Devices; Version 3.0 specification.

The following blocks can be implemented in field units according to the Profile 3.0:
- Physical Block:
  The physical block contains the unit-specific features of the unit.
- Transducer Block (transmission block):
  One or more transducer blocks contain all measurement principle and unit specific parameters of the unit. The measurement principle (e.g. temperature) according to the PROFIBUS specification is contained in the transducer blocks.
- Function Block:
  One or more function blocks contain the automation function of the unit. One can distinguish between different function blocks, e.g. analogue input block, analogue output, etc. Each of these function blocks is used for different applications.

5.2.3 System integration

On PROFIBUS-PA® field units measured values and status are generally transmitted in 5 bytes. One measurement unit with a number of process variables means that more bytes are transmitted.
In order to bind the field unit into the bus system the PROFIBUS-PA® system requires a description of the unit parameters such as output data, input data, data format, data quantity and the transmission speed supported.
This data is contained in a Device Data Base (GSD-File), that is made available to the PROFIBUS-PA® master during the commissioning of the communications system. Additionally unit bitmaps, which appear as symbols in the network, can also be included. Using the profile 3.0 Device Data Base (GSD-File) it is possible to exchange field units from different manufacturers without the need for new planning.

Generally there are two possible types of GSD using profile 3.0:
- **Manufacturers specific GSD:** With this GSD, unlimited functionality of the field unit is guaranteed. Unit-specific process parameters and functions are therefore available.
- **Profile GSD:** Differentiates itself in the number of the AI (analogue input) blocks and the measurement principles. As long as a plant has been planned using the profile GSD's, an exchange of units from different manufacturers can be made.

Note!
Before planning it must be decided with which GSD the plant is to be operated. It is possible to change this setting using a class 2 master.

Factory default set-up: Manufacturer specific GSD.

The TMT184 temperature head transmitter supports the GSD's with the ident numbers in the following table:

Unit name	Manufacturer spec. ID-no.	Profile 3.0 ID-no.	Manufacturer spec. GSD
TMT 184 PROFIBUS-PA® (IEC 61158-2)	1523 (Hex)	9700 (Hex)	EH3_1523.gsd EH3X1523.gsd
	Profile 3.0 GSD	Type-File	**Bitmaps**
	PA039700.gsd	EH31523x.200	EH1523_d.bmp EH1523_n.bmp EH1523_s.bmp

Each unit receives an identification number (ID-no.) from the Profibus users organisation (PNO). From this, the name of the Device Data Base (GSD) is formed. For Endress+Hauser this ID no. starts with the manufacturer's identification 15xx. In order to gain a better allocation and individuality to the respective GSD, the GSD names (without the type files) at Endress+Hauser are as follows:

- **EH3_15xx:**
  EH= Endress+Hauser,
  3= Profile 3.0,
  _= Standard identification and
  15xx= ID-no.

- **EH3x15xx:**
  EH= Endress+Hauser,
  3 = Profile 3.0,
  x = Expanded identification and
  15xx= ID-no.

The GSD files of all Endress+Hauser files can be requested under:
- Internet: Endress+Hauser
  www.endress.com → (Products → Process Solutions → PROFIBUS → GSD files)
- Internet: PNO
  www.profibus.com (GSD library)
- On CD ROM from Endress+Hauser: **Order number** 500 97200

Content structure of the Endress+Hauser GSD files
For the Endress+Hauser field transmitters using the PROFIBUS interface all data required for projecting is contained in one file. After unpacking, this file is created using the following structure:
The identification revision #xx stands for a specific unit version. In the directory BMP unit specific bitmaps can be found, these can be used independently from the projecting software.

The GSD files are saved in the GSD folder in the subdirectories extended and standard. Information for using the field transmitter and the dependency in the unit software is stored in the folder Info. Please carefully read these notes before projecting. The files with the suffix .200 can be found in the folder TypDat.

Standard and extended formats
There are GSD files, whose module is transmitted using an extended ident. no. (e.g. 0x42, 0x84, 0x08, 0x05). These GSD files can be found in the folder Extended. Furthermore the GSD files with a standard ident. (e.g. 0x94) can be found in the folder Standard. When integrating field transmitters the GSD files with an extended ident. should always be used first. If the integration fails, then the standard GSD should be used. This difference is the result of a specific implementation in the Master system.

Contents of the download file from the Internet and the CD-ROM
- All Endress+Hauser GSD files
- Endress+Hauser type files
- Endress+Hauser Bitmap files
- Further helpful information to the units

Working with the GSD/Type files
The GSD files can, dependent on the software being used, either be copied into the programme specific directory or read into the projecting software data bank using an import function.

Example 1
The directory...\ siemens \ step7 \ s7data \ gsd can be used for the Siemens STEP 7 of the Siemens SPS S7-300 / 400 projecting software.
Included in the GSD files are also the Bitmap files. The Bitmap files can be used to help in displaying the measurement points graphically. The Bitmap files must be stored in the directory ...\ siemens \ step7 \ s7data \ nsbmp.

Example 2
If a Siemens S5 is being used and the PROFIBUS-DP network is projected using the COM ET 200 projecting software, then the Type files will be required (x.200 files).

If there are questions regarding other projecting software types, please refer to the PLC manufacturer for the correct directory to be used.

Compatibility of Profile version 2.0 and 3.0 units
One plant can use both Profile 2.0 and 3.0 units using different GSD and a DP master, this is possible because the cyclic data for the automation system are compatible in both profile versions.

Note!
It is generally possible to exchange similar units with profile version 2.0 with units using Profile version 3.0 without exchanging the new projecting.

Note!
An exchange of Endress+Hauser TMD 834 temperature head transmitters for the newer TMT 184 is possible even though the unit is different in both name and Ident. no.
The TMT 184 is accepted as an exchangeable unit, if in the E+H unit matrix of the TMT 184 address V6H0 the change to 'MANUFACT V2.0' is activated (→ chap. 5.2.7).
The TMT 184 the operates as a replacement TMD 834 with Profile V2.0.

5.2.4 Cyclic data exchange

In the PROFIBUS-PA® the cyclic transmission of the analogue values to the automation system is done in 5 byte long blocks. The measured value is displayed in the first 4 bytes in a floating decimal point format to the IEEE 754 standard (see IEEE floating decimal point).The 5th Byte contains status information to the measured value, this is then implemented to the Profile 3.0 specification. (page 61)

IEEE floating decimal point number
Converting a Hex value into an IEEE floating decimal point number for measured value recording. The measured values are displayed in an IEEE-754 number format as follows and transmitted to the master class 1.

Byte n			Byte n+1			Byte n+2		Byte n+3	
Bit 7	Bit 6	Bit 0	Bit 7	Bit 6	Bit 0	Bit 7	Bit 0	Bit 7	Bit 0
VZ	2^7 2^6 2^5 2^4 2^3 2^2 2^1		2^0	2^{-1} 2^{-2} 2^{-3} 2^{-4} 2^{-5} 2^{-6} 2^{-7}		2^{-8} 2^{-9} 2^{-10} 2^{-11} 2^{-12} 2^{-13} 2^{-14} 2^{-15}		2^{-16} bis 2^{-23}	
	Exponent			**Mantissa**		**Mantissa**		**Mantissa**	

Formula value = $(-1)^{VZ} * 2^{(Exponent\ -127)} * (1 + Mantissa)$
Example: 40 F0 00 00 hex = 0100 0000 1111 0000 0000 0000 0000 0000 binary
Value = $(-1)^0 * 2^{(129-127)} * (1 + 2^{-1} + 2^{-2} + 2^{-3})$
$= 1 * 2^2 * (1 + 0.5 + 0.25 + 0.125)$
$= 1 * 4 * 1.875 = 7.5$
The process temperature is transmitted in a cyclic data exchange from the TMT 184 temperature head transmitter.

TMT 184 ---> Automation system

Input byte	Process parameter	Access mode	Comment/Data format	Factory default units
0, 1, 2, 3	Temperature	Read	32 Bit floating decimal point (IEEE-754)	° C
4	Status temperature	Read	page 61	-

Output data
Display value
The display value offers the possibility to transmit a value calculated in the automation system directly to the head transmitter. This measured value is a purely display value that can be displayed, for example, using an RID 261 PROFIBUS-PA display. The display value contains a 4 byte measured value an 1 byte status value.

Automation system ---> TMT 184 (display value)

Output Byte	Process parameter	Access mode	Comments/Data format
0, 1, 2, 3	Display value	Write	32 Bit floating decimal point number (IEEE-754)
4	Status display value	Write	-

Generally the projecting of a PROFIBUS-DP/-PA® system is as follows:

1. The unit to be set up (TMT 184) is connected to the PROFIBUS-PA® network using the GSD files in the configuration programme of the automation system. The values that are required can be set up offline in the projecting software.
2. The operator programme of the automation system should now be set up. The in and output data is controlled and fixed using the operator programme. Here the measured values can be found in order to be able to analyse them. If the automation system does not support the IEEE-754 floating decimal point format then an addition measured value conversion block must be used. Dependent on the type of data management used by the automation system (Little-Endian-Format or Big-Endian-Format), a change in the byte sequence may be needed (byte-swapping).
3. Once the projecting has been completed this is transmitted to the automation system as a binary file.
4. The system can be started once the projecting sequence has been completed. The automation system connects to the projected units. Now the process relevant unit parameters can be set up using a class 2 master, e.g. using Commuwin II.

Status code

Status codes, supported by the AI block (analogue input).
Coding for the status in compliance with the PROFIBUS Profiles 3.0 'PROFIBUS-PA Profile for Process Control Devices - General Requirements' V 3.0:

Status Code	Meaning	Unit condition	Limits
0x1C	OUT_OF SERVICE	BAD	OK
0x1D	OUT_OF SERVICE	BAD	LOW_LIM
0x1E	OUT_OF SERVICE	BAD	HIG_LIM
0x1F	OUT_OF SERVICE	BAD	CONST
0x0C	DEVICE_FAILURE	BAD	OK
0x0D	DEVICE_FAILURE	BAD	LOW_LIM
0x0E	DEVICE_FAILURE	BAD	HIG_LIM
0x0F	DEVICE_FAILURE	BAD	CONST
0x44	LAST_USABLE_VALUE	UNCERTAIN	OK
0x45	LAST_USABLE_VALUE	UNCERTAIN	LOW_LIM
0x46	LAST_USABLE_VALUE	UNCERTAIN	HIG_LIM
0x47	LAST_USABLE_VALUE	UNCERTAIN	CONST
0x48	SUBSTITUTE_SET	UNCERTAIN	OK
0x49	SUBSTITUTE_SET	UNCERTAIN	LOW_LIM
0x4A	SUBSTITUTE_SET	UNCERTAIN	HIG_LIM
0x4B	SUBSTITUTE_SET	UNCERTAIN	CONST
0x4C	INITIAL_VALUE	UNCERTAIN	OK
0x4D	INITIAL_VALUE	UNCERTAIN	LOW_LIM
0x4E	INITIAL_VALUE	UNCERTAIN	HIG_LIM
0x4F	INITIAL_VALUE	UNCERTAIN	CONST

Status Code	Meaning	Unit condition	Limits
0x80 0x81 0x82 0x83	NC_OK NC_OK NC_OK NC_OK	GOOD GOOD GOOD GOOD	OK LOW_LIM HIG_LIM CONST
0x84 0x85 0x86 0x87	NC_OK_UPDATE_EVENT NC_OK_UPDATE_EVENT NC_OK_UPDATE_EVENT NC_OK_UPDATE_EVENT	GOOD GOOD GOOD GOOD	OK LOW_LIM HIG_LIM CONST
0x8C 0x8D 0x8E 0x8F	NC_OK_CRIT_ALARM NC_OK_CRIT_ALARM NC_OK_CRIT_ALARM NC_OK_CRIT_ALARM	GOOD GOOD GOOD GOOD	OK LOW_LIM HIG_LIM CONST
0x88 0x89 0x8A 0x8B	NC_OK_ADVISORY_AL NC_OK_ADVISORY_AL NC_OK_ADVISORY_AL NC_OK_ADVISORY_AL	GOOD GOOD GOOD GOOD	OK LOW_LIM HIG_LIM CONST

5.2.5 Acyclic data transfer

The acyclic data transmission is for transmitting parameters during commissioning, for maintenance or for display of other measurements that are not contained in the cyclic data traffic.

Generally a difference is made between class 1 and class 2 master connections. A class 2 master is allowed on the TMT 184 temperature head transmitter. When reading parameters by a class 2 master a request telegram is sent from the class 2 master to the field unit using the field unit address, slot and index as well as the expected data length. The field unit answers using the requested data set, if this exists and has the correct length (in byte).
When writing parameters using a class 2 master the slot and index, lengths (byte) and data set are transmitted as well as the field unit address. The slave acknowledges this write command once it is finished. Using a class 2 master access to the blocks, that are indicated in the following figure is permitted.
The parameters, that are operated using the E+H operation programme (Commuwin II) are displayed in the form of a matrix (page 64).

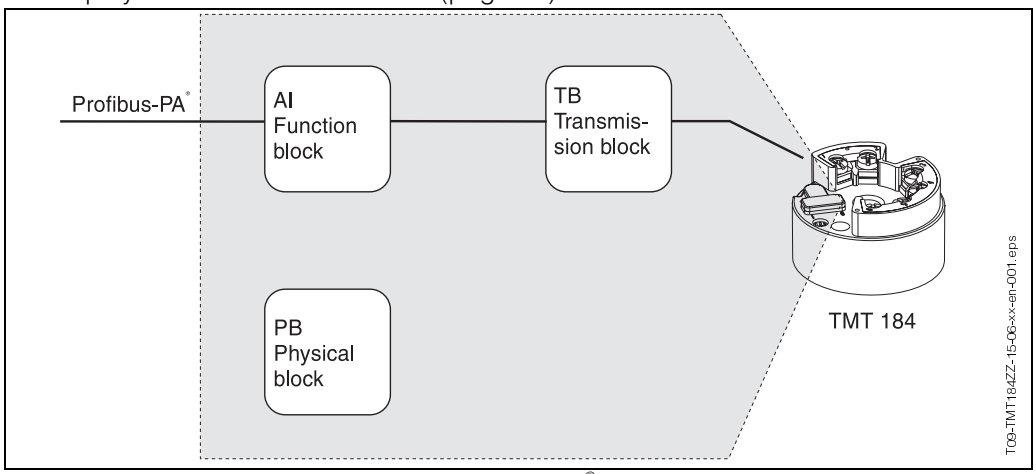

Fig. 5.3: Function block model of the TMT 184 PROFIBUS-PA®

5.2.11 TMT 184 Slot / Index Lists

Note!
Detailed information on the following tables can be found under 'PROFIBUS-PA Profile for Process Control Devices; Version 3.0'.

Index	Parameter	Object type	Data type	Store	Size (in bytes)	Acc.	Parameter usage/ Type of transport	Default values
0	**Directory Header**	Array	Unsigned16	Cst	12	r	a	-
1	**Composite list directory entry/ Composite directory entries**	Array	Unsigned16	Cst	24	r	a	-
2-8	**Directory_continuous**	Array	Unsigned16	Cst	*	r	a	-
16	**Analog Input Block Object**	Record	DS-32	Cst	20	r	C/a	-
17	**ST_REV**	Simple	Unsigned16	N	2	r	C/a	0
18	**TAG_DESC**	Simple	OctetString	S	32	r,w	C/a	"
19	**STRATEGY**	Simple	Unsigned16	S	2	r,w	C/a	0
20	**ALERT_KEY**	Simple	Unsigned8	S	1	r,w	C/a	0
21	**TARGET_MODE**	Simple	Unsigned8	S	1	r,w	C/a	-
22	**MODE_BLK**	Record	DS-37	D	3	r	C/a	block-specific
23	**ALARM_SUM**	Record	DS-42	D	8	r	C/a	0,0,0,0
24	**BATCH**	Struct.	DS-67	S	10	r,w	C/a	0,0,0,0
26	**OUT**	Record	DS-33	D	5	r	O/cyc	measured of the variable, state
27	**PV_SCALE**	Array	Float	S	8	r,w	C/a	0,100
28	**OUT_SCALE**	Record	DS-36	S	11	r,w	C/a	0,100, °C,2
29	**LIN_TYPE**	Simple	Unsigned8	S	1	r,w	C/a	0
30	**CHANNEL**	Simple	Unsigned16	S	2	r,w	C/a	-
32	**PV_FTIME**	Simple	Float	N	4	r,w	C/a	0
33	**FSAFE_TYPE**	Simple	Unsigned8	S	1	r,w	C/a	1
34	**FSAFE_VALID**	Simple	Float	S	4	r,w	C/a	-
35	**ALARM_HYS**	Simple	Float	S	4	r,w	C/a	0,5% of range

37	**HI_HI_LIM**	Simple	Float	S	4	r,w	C/a	Max value
39	**HI_LIM**	Simple	Float	S	4	r,w	C/a	Max value
41	**LO_LIM**	Simple	Float	S	4	r,w	C/a	Min value
43	**LO_LO_LIM**	Simple	Float	S	4	r,w	C/a	Min value
46	**HI_HI_ALM**	Record	DS-39	D	16	r	C/a	0
47	**HI_ALM**	Record	DS-39	D	16	r	C/a	0
48	**LO_ALM**	Record	DS-39	D	16	r	C/a	0
49	**LO_LO_ALM**	Record	DS-39	D	16	r	C/a	0
50	**SIMULATE**	Record	DS-50	N	6	r,w	C/a	Disable
51	**OUT_UNIT_TEXT**	Simple	OctetString	S	16	r,w	C/a	-
67	**Physical Block Object**	Record	DS-32	Cst	20	r	C/a	-
68	**ST_REV**	Simple	Unsigned16	N	2	r	C/a	0
69	**TAG_DESC**	Simple	OctetString	S	32	r,w	C/a	"
70	**STRATEGY**	Simple	Unsigned16	S	2	r,w	C/a	0
71	**ALERT_KEY**	Simple	Unsigned8	S	1	r,w	C/a	0
72	**TARGET_MODE**	Simple	Unsigned8	S	1	r,w	C/a	-
73	**MODE_BLK**	Record	DS-37	D	3	r	C/a	block-specific
74	**ALARM_SUM**	Record	DS-42	D	8	r	C/a	0,0,0,0
75	**SOFTWARE_REVISION**	Simple	VisibleString	Cst	16	r	C/a	-
76	**HARDWARE_REVISION**	Simple	VisibleString	Cst	16	r	C/a	-
77	**DEVICE_MAN_ID**	Simple	Unsigned16	Cst	2	r, w(k)	C/a	-
78	**DEVICE_ID**	Simple	VisibleString	Cst	16	r, w(k)	C/a	-
79	**DEVICE_SER_Num**	Simple	VisibleString	Cst	16	r, w(k)	C/a	-
80	**DIAGNOSIS**	Simple	Oct.str byt4, MSB=1 more diag avail.	D	4	r	C/a	-
81	**DIAGNOSIS_EXTENSION**	Simple	Octetstring	D	6	r	C/a	-
82	**DIAGNOSIS_MASK**	Simple	Octetstring	Cst	4	r	C/a	-
83	**DIAGNOSIS_MASK_EXTENSION**	Simple	Octetstring	Cst	6	r	C/a	-
84	**DEVICE_CERTIFICATION**	Simple	VisibleString	Cst	32	r	C/a	-

85	**WRITE_LOCKING**	Simple	Unsigned16	N	2	r,w	C/a	-
86	**FACTORY_RESET**	Simple	Unsigned16	S	2	r,w	C/a	-
87	**DESCRIPTOR**	Simple	OctetString	S	32	r,w	C/a	-
88	**DEVICE_MESSAGE**	Simple	OctetString	S	32	r,w	C/a	-
89	**DEVICE_INSTAL_DATE**	Simple	OctetString	S	16	r,w	C/a	-
91	**IDENT_NUMBER_SELECTOR**	Simple	Unsigned8	S	1	r,w	C/a	-
105	**Actual Error**	Simple	Unsigned16	D	2	r		
106	**Last Error**	Simple	Unsigned16	D/S	2	r,w		
107	**UpDownFeaturesSupported**	Simple	OctetString	C	1	r		
108	**UpDownCtrlParameter**	Simple	Unsigned8	D	1	w		
109	**UpDownParameter**	Record	UpDown Data	D	20	r,w		
110	**Device Bus Address**	Simple	Int8	D/S	1	r		
111	**Device and Software Number**	Simple	Unsigned16	C	2	r		
112	**Set Unit to Bus**	Simple	Unsigned8	V	1	w		
113	**Local Display Input**	Record	LocalDispVal	D	6	r,w		
121	**Ident No.**	Simple	Unsigned16	D	2	r		
122	**DP-Status**	Simple	Unsigned8	D	1	r		
128	**Temperature Transducer Block Object**	Record	DS-32	Cst	20	r	C/a	-
129	**ST_REV**	Simple	Unsigned16	N	2	r	C/a	0
130	**TAG_DESC**	Simple	OctetString	S	32	r,w	C/a	‘‘
131	**STRATEGY**	Simple	Unsigned16	S	2	r,w	C/a	0
132	**ALERT_KEY**	Simple	Unsigned8	S	1	r,w	C/a	0
133	**TARGET_MODE**	Simple	Unsigned8	S	1	r,w	C/a	-
134	**MODE_BLK**	Record	DS-37	D	3	r	C/a	block-specific
135	**ALARM_SUM**	Record	DS-42	D	8	r	C/a	0,0,0,0
136	**PRIMARY_VALUE**	Simple	DS-33	D	5	r	C/a	
137	**PRIMARY_VALUE_UNIT**	Simple	Unsigned16	S	2	r,w	C/a	
138	**SECONDARY_VALUE_1**	Simple	DS-33	D	5	r	C/a	
140	**SENSOR_MEAS_TYPE**	Simple	Unsigned8	S	1	r,w	C/a	0
141	**INPUT_RANGE**	Simple	Unsigned8	S	1	r,w	C/a	

142	**LIN_TYPE**	Simple	Unsigned8	S	1	r,w	C/a	
147	**BIAS_1**	Simple	Float	S	4	r,w	C/a	0.0
149	**UPPER_SENSOR_LIMIT**	Simple	Float	N	4	r	C/a	
150	**LOWER_SENSOR_LIMIT**	Simple	Float	N	4	r	C/a	
152	**INPUT_FAULT_GEN**	Simple	Unsigned8	D	1	r	C/a	
153	**INPUT_FAULT_1**	Simple	Unsigned8	D	1	r	C/a	
161	**RJ_TEMP**	Simple	Float	D	4	r	C/a	
162	**RJ_TYPE**	Simple	Unsigned8	S	1	r,w	C/a	
163	**EXTERNAL_RJ_VALUE**	Simple	Float	S	4	r,w	C/a	
164	**SENSOR_CONNECTION**	Simple	Unsigned8	S	1	r,w	C/a	
165	**COMP_WIRE1**	Simple	Float	S	4	r,w	C/a	
199	**Viewobject of Analog Input Function Block**				18	r		
200	**Viewobject of Physical Block**				17	r		
201	**Viewobject of Temperature Transducer Block**				20	r		

6 Commissioning

6.1 Installation and function checks

Installation and function checks

Make sure that all checks have been made before the unit is operational:
- See »Connection control« on page 55.

Note!
- All function technical data of the PROFIBUS interface to the IEC 61158-2 must be adhered to (FISCO_model).
- Monitor the Bus voltage using a standard multi-meter. This should lie from 9 to 32 volts with a current of 10 ± 1 mA at the unit.

6.2 Commissioning

Commissioning

The head transmitter is operational once the power supply is active.

6.2.1 Quick Setup

A Quick Setup of the PROFIBUS-PA® head transmitters is possible using the E+H operating programme Commuwin II and the E+H unit matrix (→ chap. 5.2.7). The function description of the parameters can be found in the matrix description in chapter »Unit matrix PROFIBUS-PA® temperature head transmitter« on page 64.

6.2.2 Set-up with PROFIBUS

Set-up unit address
- Addressing:
  The address of a PROFIBUS-PA® unit must always be set up. Valid unit addresses lie in the range 0 to 125. In a PROFIBUS-PA® network an address can only be allocated once. If the address is set up incorrectly the measurement unit will not be recognised by the master. The address 126 is to be used for initial commissioning and service reasons.
- Delivery condition:
  All units are delivered with the address 126 and software addressing.
- Software addressing PROFIBUS-PA® using the DPV1 server of Commuwin II. The changeover of the software addressing is done via this server.

Note!
Connection between the operating programme Commuwin II must be disconnected using the menu **Connection → disconnect** before changing the software addressing!

Addressing PROFIBUS-PA® using the DIP switch (optional, s. fig. 6.1)
Open the cover to the DIP switches on the head transmitter. Using a pointed object (small screwdriver) set the position of the mini switches to the required address. Once this has been done close the cover to the DIP switches..

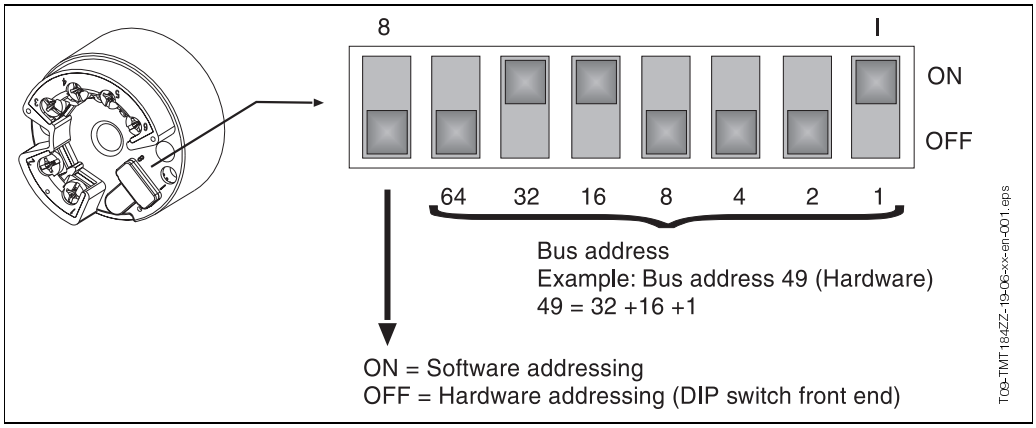

Fig. 6.1: Addressing using the DIP switches

7 Maintenance

Maintenance

The head transmitter is maintenance-free.

8 Accessories

Accessories

PC software COMMUWIN II, Proficard or Profiboard, Segment coupler.
Please contact your local supplier when ordering (e.g. accessories and spare parts)!

9 Troubleshooting

9.1 Troubleshooting instructions

Troubleshooting instructions

If faults occur after commissioning or during measurement, always start any troubleshooting sequence using the following check. The user is led towards the possible fault cause and its rectification via question and answer.