# KNX W sl

## Wind Sensor

Item number 70158

elsner elektronik

## Installation and Adjustment

Installation, inspection, commissioning and troubleshooting of the device must only be carried out by a competent electrician.

This manual is amended periodically and will be brought into line with new software releases. The change status (software version and date) can be found in the contents footer. If you have a device with a later software version, please check **www.elsner-elektronik.de** in the menu area "Service" to find out whether a more up-to-date version of the manual is available.

## Clarification of signs used in this manual

Safety advice.

Safety advice for working on electrical connections, components, etc.

**DANGER!** ... indicates an immediately hazardous situation which will lead to death or severe injuries if it is not avoided.

**WARNING!** ... indicates a potentially hazardous situation which may lead to death or severe injuries if it is not avoided.

**CAUTION!** ... indicates a potentially hazardous situation which may lead to trivial or minor injuries if it is not avoided.


**ATTENTION!** ... indicates a situation which may lead to damage to property if it is not avoided.

ETS In the ETS tables, the parameter default settings are marked by underlining.

# 1. Description

The **Wind Sensor KNX W sl** for the KNX building bus system measures wind speed. The wind value can be used for the control of limit dependent switching outputs. States can be linked via AND logic gates and OR logic gates. Multi-function modules change input data as required by means of calculations, querying a condition, or converting the data point type.

The compact housing of the **KNX W sl** accommodates the sensors, evaluation circuits and bus-coupling electronics.

***Functions:***

- **Wind measurement**: The wind strength is measured electronically and thus noiselessly and reliably, even during hail, snow and sub-zero temperatures. Even turbulent air and rising winds in the vicinity of the device are recorded
- **Switching outputs** for all measured and computed values. Threshold values can be adjusted per parameter or via communication objects
- **8 AND and 8 OR logic gates**, each with 4 inputs. All switching events as well as 16 logic inputs (in the form of communications objects) can be used as inputs for the logic gates. The output of each gate can be configured optionally as 1-bit or 2 x 8-bit
- **8 multi-function modules** (computers) for changing the input data by calculations, by querying a condition or by converting the data point type

Configuration is made using the KNX software ETS. The **product file** can be downloaded from the Elsner Elektronik website on **www.elsner-elektronik.de** in the "Service" menu.

### 1.0.1. Deliverables

- Sensor
- Connection cable approx. 3 m, with plug
- Surface-mounted junction box (IP 55)
- Worm drive hose clip for pole mounting (Ø 40-60 mm)
- 4×50 mm stainless steel roundhead screws and 6×30 mm dowels for wall mounting. Use fixing materials that are suitable for the base!

## 1.1. Technical specification

| | |
|---|---|
| Housing | Plastic |
| Colour | White / Translucent |
| Assembly | Surface mount |
| Protection category | IP 44 |
| Dimensions | approx. 62 × 71 × 156 (W × H × D, mm) |
| Weight | Sensor with mount, approx. 90 g,<br>Total weight including accessories, approx.280 g |
| Ambient temperature | Operation -30…+50°C, storage -30…+70°C |

| | |
|---|---|
| Auxiliary supply | 12…40 V DC, 12…28 V AC.<br>An appropriate power supply unit can be purchased from Elsner Elektronik. |
| Auxiliary current | at 12V DC: max. 185 mA<br>at 24V DC: max. 90 mA<br>at 24V AC: max. 82 mA |
| Bus current | max. 10 mA |
| Data output | KNX +/- |
| BCU type | Integrated microcontroller |
| PEI type | 0 |
| Group addresses | max. 2000 |
| Assignments | max. 2000 |
| Communication objects | 176 |
| Wind sensor: | |
| Measurement range | 0 m/s ... 35 m/s |
| Resolution | 0.1 m/s |
| Accuracy | ±15% of the measurement value when incoming flow is 45°…315°<br>(Frontal incoming flow corresponds to 180°) |

The product conforms with the provisions of EU directives.

# 2. Installation and start-up

## 2.1. Installation notes

Installation, testing, operational start-up and troubleshooting should only be performed by an electrician.

**CAUTION!**
**Live voltage!**
There are unprotected live components inside the device.

- National legal regulations are to be followed.
- Ensure that all lines to be assembled are free of voltage and take precautions against accidental switching on.
- Do not use the device if it is damaged.
- Take the device or system out of service and secure it against unintentional use, if it can be assumed, that risk-free operation is no longer guaranteed.

The device is only to be used for its intended purpose. Any improper modification or failure to follow the operating instructions voids any and all warranty and guarantee claims.

After unpacking the device, check it immediately for possible mechanical damage. If it has been damaged in transport, inform the supplier immediately.

The device may only be used as a fixed-site installation; that means only when assembled and after conclusion of all installation and operational start-up tasks and only in the surroundings designated for it.

Elsner Elektronik is not liable for any changes in norms and standards which may occur after publication of these operating instructions.

## 2.2. Installation location

Select an installation position on the building where the sensor can measure wind without hindrance. Please ensure that the extended awning is protected from the wind.

At least 60 cm of clearance must be left around the device. This facilitates correct wind speed measurement without eddies. At the same time, this prevents spray (raindrops hitting the device) or snow (snow penetration) from impairing the measurement. The wind sensor must not come into contact with water. This also prevents birds from biting it.

The mounting position must be selected so that the wind sensor cannot be touched by persons.

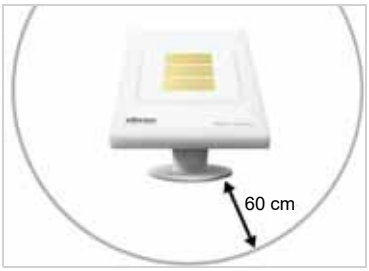


*Fig. 1*
*There must be at least 60 cm clearance to other elements (structures, construction parts, etc.) below, to the sides and in front of the device.*

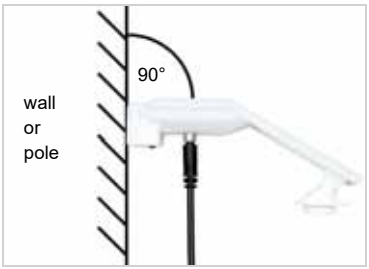


*Fig. 2*
*The device must be attached to a vertical wall (or a pole).*


*Fig. 3*
*The device must be mounted in the horizontal (transverse) direction.*

### 2.2.1. Sensor position


*Fig. 4*

*1 Wind measuring element*

**ATTENTION!**
Sensitive wind sensor.
- Remove the protective transport sticker after installation.
- Do not touch the sensor on the wind measuring element (below, countersunk).

## 2.3. Sensor assembly

### 2.3.1. Attach mount

First, assemble the mount for wall/pole mounting. Release the screw joint of the mount with a cross-headed screwdriver.

### *Wall installation*

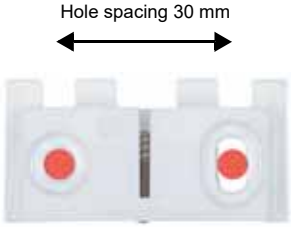


*Fig. 5 Front view*

Use two screws to attach the mount to the wall. Use the fastening material (dowels, screws) that is suitable for the base.

Make sure that the arrows are pointing upward.

### *Pole installation*

The device is installed on the pole with the enclosed clamp.

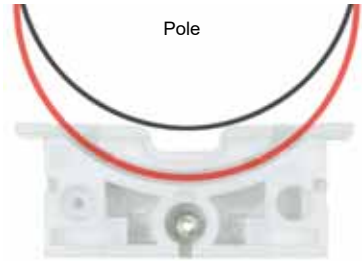


*Fig. 6 Bottom view*

Insert the clamp in the mount through the recess. Tighten the clamp on the pole.

Make sure that the arrows are pointing upward.

### 2.3.2. Attaching and connecting the device


*Fig. 7*

1. Slide the device onto the mounting from above.
2. Tighten the screw of the mount to secure the device.
3. Screw the M8 connectors of the connection cable onto the connection socket on the bottom side of the device.

Connect the loose end of the connection cable to the KNX bus and auxiliary voltage. Use the connection sockets and clips included for this purpose.

| *KNX bus:* | *Auxiliary voltage:* |
|---|---|
| + *Red* | + *Yellow* |
| - *Black* | - *White* |

## 2.4. Instructions for assembly and initial start-up

Remove all transport protection stickers present after installation.

The wind measurement value and thus also all wind switching outputs cannot be output until 35 seconds after the power is turned on.

After the auxiliary voltage has been applied, the device will enter an initialisation phase lasting a few seconds. During this phase no information can be received or sent via the bus.

# 3. Addressing the equipment

The equipment is delivered ex works with the bus address 15.15.255. You program a different address in the ETS by overwriting the address 15.15.255 or teach the device using the programming button.

The programming button can be reached through the opening on the underside of the housing; it is recessed by approx. 15 mm. Use a thin object to reach the key, e. g. a 1.5 mm² wire.

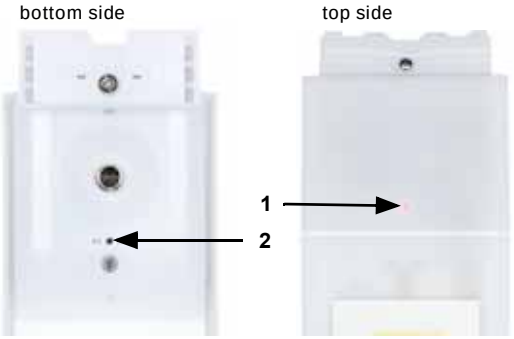


*Fig. 8*
*1 Programming LED (under the semi-transparent lid)*
*2 Programming button for teaching the device*

# 4. Maintenance

**WARNING!**
**Risk of injury caused by components moved automatically!**
The automatic control can start system components and place people in danger (e.g. moving windows/awnings if a rain/wind alarm has been triggered while cleaning).
- Always isolate the device from the mains for servicing and cleaning.

The device must regularly be checked for dirt twice a year and cleaned if necessary. In case of severe dirt, the sensor may not work properly anymore.

**ATTENTION**
The device can be damaged if water penetrates the housing.
- Do not clean with high pressure cleaners or steam jets.

# 5. Transfer protocol

***Units:***

*Wind in metres per second*

## 5.1. List of all communications objects

***Abbreviation flags:***

*C Communication*
*R Read*
*W Write*
*T Transmit*
*U Update*

| No. | Text | Function | Flags | DPT type | Size |
|---|---|---|---|---|---|
| 1 | Software version | Output | R-CT | [217.1] DPT_Version | 2 bytes |
| 21 | Signal LED object 1s cycle | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 22 | Signal LED object 4s cycle | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 271 | Wind sensor: Malfunction | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 272 | Wind sensor: Measurement [m/s] | Output | R-CT | [9.5] DPT_Value_Wsp | 2 bytes |
| 273 | Wind sensor: Measurement [Beaufort] | Output | R-CT | [20.014] DPT_Beaufort_Wind_Force_Scale | 1 byte |
| 274 | Wind sensor: Max. query measurement | Input | -WC- | [1.017] DPT_Trigger | 1 bit |
| 275 | Wind sensor: Maximum measurement [m/s] | Output | R-CT | [9.5] DPT_Value_Wsp | 2 bytes |
| 276 | Wind sensor: Maximum measurement [Beaufort] | Output | R-CT | [20.014] DPT_Beaufort_Wind_Force_Scale | 1 byte |
| 277 | Wind sensor: Max. reset measurement | Input | -WC- | [1.017] DPT_Trigger | 1 bit |
| 281 | Wind threshold value 1: Absolute value | Input/ Output | RWCT | [9.5] DPT_Value_Wsp | 2 bytes |
| 282 | Wind threshold value 1: (1:+ \| 0:-) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 283 | Wind threshold value 1: Delay from 0 to 1 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 284 | Wind threshold value 1: Delay from 1 to 0 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 285 | Wind threshold value 1: Switching output | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 286 | Wind threshold value 1: Switching output block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 287 | Wind threshold value 2: Absolute value | Input/ Output | RWCT | [9.5] DPT_Value_Wsp | 2 bytes |
| 288 | Wind threshold value 2: (1:+ \| 0:-) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 289 | Wind threshold value 2: Delay from 0 to 1 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |

| No. | Text | Function | Flags | DPT type | Size |
|---|---|---|---|---|---|
| 290 | Wind threshold value 2: Delay from 1 to 0 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 291 | Wind threshold value 2: Switching output | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 292 | Wind threshold value 2: Switching output block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 293 | Wind threshold value 3: Absolute value | Input/ Output | RWCT | [9.5] DPT_Value_Wsp | 2 bytes |
| 294 | Wind threshold value 3: (1:+ \| 0:-) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 295 | Wind threshold value 3: Delay from 0 to 1 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 296 | Wind threshold value 3: Delay from 1 to 0 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 297 | Wind threshold value 3: Switching output | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 298 | Wind threshold value 3: Switching output block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 299 | Wind threshold value 4: Absolute value | Input/ Output | RWCT | [9.5] DPT_Value_Wsp | 2 bytes |
| 300 | Wind threshold value 4: (1:+ \| 0:-) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 301 | Wind threshold value 4: Delay from 0 to 1 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 302 | Wind threshold value 4: Delay from 1 to 0 | Input | -WC- | [7.5] DPT_TimePeriodSec | 2 bytes |
| 303 | Wind threshold value 4: Switching output | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 304 | Wind threshold value 4: Switching output block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1141 | Computer 1: Input I1 | Input | RWCT | | 4 bytes |
| 1142 | Computer 1: Input I2 | Input | RWCT | | 4 bytes |
| 1143 | Computer 1: Input I3 | Input | RWCT | | 4 bytes |
| 1144 | Computer 1: Output O1 | Output | R-CT | | 4 bytes |
| 1145 | Computer 1: Output O2 | Output | R-CT | | 4 bytes |
| 1146 | Computer 1: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1147 | Computer 1: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1148 | Computer 1: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1149 | Computer 2: Input I1 | Input | RWCT | | 4 bytes |
| 1150 | Computer 2: Input I2 | Input | RWCT | | 4 bytes |
| 1151 | Computer 2: Input I3 | Input | RWCT | | 4 bytes |
| 1152 | Computer 2: Output O1 | Output | R-CT | | 4 bytes |
| 1153 | Computer 2: Output O2 | Output | R-CT | | 4 bytes |
| 1154 | Computer 2: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1155 | Computer 2: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1156 | Computer 2: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |

| No. | Text | Function | Flags | DPT type | Size |
|---|---|---|---|---|---|
| 1157 | Computer 3: Input I1 | Input | RWCT | | 4 bytes |
| 1158 | Computer 3: Input I2 | Input | RWCT | | 4 bytes |
| 1159 | Computer 3: Input I3 | Input | RWCT | | 4 bytes |
| 1160 | Computer 3: Output O1 | Output | R-CT | | 4 bytes |
| 1161 | Computer 3: Output O2 | Output | R-CT | | 4 bytes |
| 1162 | Computer 3: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1163 | Computer 3: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1164 | Computer 3: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1165 | Computer 4: Input I1 | Input | RWCT | | 4 bytes |
| 1166 | Computer 4: Input I2 | Input | RWCT | | 4 bytes |
| 1167 | Computer 4: Input I3 | Input | RWCT | | 4 bytes |
| 1168 | Computer 4: Output O1 | Output | R-CT | | 4 bytes |
| 1169 | Computer 4: Output O2 | Output | R-CT | | 4 bytes |
| 1170 | Computer 4: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1171 | Computer 4: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1172 | Computer 4: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1173 | Computer 5: Input I1 | Input | RWCT | | 4 bytes |
| 1174 | Computer 5: Input I2 | Input | RWCT | | 4 bytes |
| 1175 | Computer 5: Input I3 | Input | RWCT | | 4 bytes |
| 1176 | Computer 5: Output O1 | Output | R-CT | | 4 bytes |
| 1177 | Computer 5: Output O2 | Output | R-CT | | 4 bytes |
| 1178 | Computer 5: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1179 | Computer 5: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1180 | Computer 5: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1181 | Computer 6: Input I1 | Input | RWCT | | 4 bytes |
| 1182 | Computer 6: Input I2 | Input | RWCT | | 4 bytes |
| 1183 | Computer 6: Input I3 | Input | RWCT | | 4 bytes |
| 1184 | Computer 6: Output O1 | Output | R-CT | | 4 bytes |
| 1185 | Computer 6: Output O2 | Output | R-CT | | 4 bytes |
| 1186 | Computer 6: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1187 | Computer 6: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1188 | Computer 6: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1189 | Computer 7: Input I1 | Input | RWCT | | 4 bytes |
| 1190 | Computer 7: Input I2 | Input | RWCT | | 4 bytes |
| 1191 | Computer 7: Input I3 | Input | RWCT | | 4 bytes |
| 1192 | Computer 7: Output O1 | Output | R-CT | | 4 bytes |
| 1193 | Computer 7: Output O2 | Output | R-CT | | 4 bytes |
| 1194 | Computer 7: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1195 | Computer 7: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |

| No. | Text | Function | Flags | DPT type | Size |
|---|---|---|---|---|---|
| 1196 | Computer 7: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1197 | Computer 8: Input I1 | Input | RWCT | | 4 bytes |
| 1198 | Computer 8: Input I2 | Input | RWCT | | 4 bytes |
| 1199 | Computer 8: Input I3 | Input | RWCT | | 4 bytes |
| 1200 | Computer 8: Output O1 | Output | R-CT | | 4 bytes |
| 1201 | Computer 8: Output O2 | Output | R-CT | | 4 bytes |
| 1202 | Computer 8: Condition text | Output | R-CT | [16.0] DPT_String_ASCII | 14 bytes |
| 1203 | Computer 8: Monitoring status | Output | R-CT | [1.1] DPT_Switch | 1 bit |
| 1204 | Computer 8: Block (1: block) | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1391 | Logic input 1 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1392 | Logic input 2 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1393 | Logic input 3 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1394 | Logic input 4 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1395 | Logic input 5 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1396 | Logic input 6 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1397 | Logic input 7 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1398 | Logic input 8 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1399 | Logic input 9 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1400 | Logic input 10 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1401 | Logic input 11 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1402 | Logic input 12 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1403 | Logic input 13 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1404 | Logic input 14 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1405 | Logic input 15 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1406 | Logic input 16 | Input | -WC- | [1.2] DPT_Bool | 1 bit |
| 1411 | AND logic 1: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1412 | AND logic 1: 8-bit output A | Output | R-CT | | 1 byte |
| 1413 | AND logic 1: 8-bit output B | Output | R-CT | | 1 byte |
| 1414 | AND logic 1: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1415 | AND logic 2: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1416 | AND logic 2: 8-bit output A | Output | R-CT | | 1 byte |
| 1417 | AND logic 2: 8-bit output B | Output | R-CT | | 1 byte |
| 1418 | AND logic 2: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1419 | AND logic 3: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1420 | AND logic 3: 8-bit output A | Output | R-CT | | 1 byte |
| 1421 | AND logic 3: 8-bit output B | Output | R-CT | | 1 byte |
| 1422 | AND logic 3: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1423 | AND logic 4: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1424 | AND logic 4: 8-bit output A | Output | R-CT | | 1 byte |
| 1425 | AND logic 4: 8-bit output B | Output | R-CT | | 1 byte |
| 1426 | AND logic 4: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1427 | AND logic 5: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |

| No. | Text | Function | Flags | DPT type | Size |
|---|---|---|---|---|---|
| 1428 | AND logic 5: 8-bit output A | Output | R-CT | | 1 byte |
| 1429 | AND logic 5: 8-bit output B | Output | R-CT | | 1 byte |
| 1430 | AND logic 5: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1431 | AND logic 6: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1432 | AND logic 6: 8-bit output A | Output | R-CT | | 1 byte |
| 1433 | AND logic 6: 8-bit output B | Output | R-CT | | 1 byte |
| 1434 | AND logic 6: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1435 | AND logic 7: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1436 | AND logic 7: 8-bit output A | Output | R-CT | | 1 byte |
| 1437 | AND logic 7: 8-bit output B | Output | R-CT | | 1 byte |
| 1438 | AND logic 7: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1439 | AND logic 8: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1440 | AND logic 8: 8-bit output A | Output | R-CT | | 1 byte |
| 1441 | AND logic 8: 8-bit output B | Output | R-CT | | 1 byte |
| 1442 | AND logic 8: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1443 | OR logic 1: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1444 | OR logic 1: 8-bit output A | Output | R-CT | | 1 byte |
| 1445 | OR logic 1: 8-bit output B | Output | R-CT | | 1 byte |
| 1446 | OR logic 1: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1447 | OR logic 2: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1448 | OR logic 2: 8-bit output A | Output | R-CT | | 1 byte |
| 1449 | OR logic 2: 8-bit output B | Output | R-CT | | 1 byte |
| 1450 | OR logic 2: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1451 | OR logic 3: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1452 | OR logic 3: 8-bit output A | Output | R-CT | | 1 byte |
| 1453 | OR logic 3: 8-bit output B | Output | R-CT | | 1 byte |
| 1454 | OR logic 3: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1455 | OR logic 4: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1456 | OR logic 4: 8-bit output A | Output | R-CT | | 1 byte |
| 1457 | OR logic 4: 8-bit output B | Output | R-CT | | 1 byte |
| 1458 | OR logic 4: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1459 | OR logic 5: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1460 | OR logic 5: 8-bit output A | Output | R-CT | | 1 byte |
| 1461 | OR logic 5: 8-bit output B | Output | R-CT | | 1 byte |
| 1462 | OR logic 5: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1463 | OR logic 6: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1464 | OR logic 6: 8-bit output A | Output | R-CT | | 1 byte |
| 1465 | OR logic 6: 8-bit output B | Output | R-CT | | 1 byte |
| 1466 | OR logic 6: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1467 | OR logic 7: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1468 | OR logic 7: 8-bit output A | Output | R-CT | | 1 byte |
| 1469 | OR logic 7: 8-bit output B | Output | R-CT | | 1 byte |

| No. | Text | Function | Flags | DPT type | Size |
|---|---|---|---|---|---|
| 1470 | OR logic 7: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |
| 1471 | OR logic 8: 1-bit switching output | Output | R-CT | [1.2] DPT_Bool | 1 bit |
| 1472 | OR logic 8: 8-bit output A | Output | R-CT | | 1 byte |
| 1473 | OR logic 8: 8-bit output B | Output | R-CT | | 1 byte |
| 1474 | OR logic 8: Block | Input | -WC- | [1.1] DPT_Switch | 1 bit |

# 6. Parameter setting

### 6.0.1. Behaviour on power failure/power restoration

***Behaviour on bus or auxiliary power failure***

The device sends nothing.

***Behaviour on bus or auxiliary voltage restoration and following programming or reset***

The device sends all measurement values as well as switching and status outputs according to their send pattern set in the parameters with the delays established in the "General settings" parameter block. The "Software version" communications object is sent once after 5 seconds.

### 6.0.2. Storage of threshold values

For threshold values that are specified via a communication object, a starting value must be entered for the first commissioning. It is valid until the first communication of a new threshold value.

After this, a threshold value once set per parameter or via a communication object is retained until a new threshold value is sent via a communication object. The last threshold value set by communication object is saved in the device, so that it is retained during a power outage and is available once again when power is restored.

### 6.0.3. Malfunction objects

Malfunction objects are sent after every reset and, additionally, after changes (i.e. at the beginning and end of a malfunction).

## 6.1. General settings

Set basic characteristics of data transfer. A different transmission delay prevents an overload of the bus shortly after the reset.

| Transmission delay after reset/restoration of bus for: | |
|---|---|
| Measured values | 5 ... 300 seconds |
| Threshold values and switching outputs | 5 ... 300 seconds |
| Computer objects | 5 ... 300 seconds |

| | |
|---|---|
| Logic objects | 5 ... 300 seconds |
| Maximum telegram quota | 1 • 2 • 5 • 10 • 20 • 50 Telegrams per sec. |

## 6.2. Wind measurement

If necessary, activate the wind malfunction object. Specify whether the measurement should also be output in Beaufort.

| | |
|---|---|
| Use malfunction object | No • Yes |
| Measured value additionally output in the Beaufort scale | No • Yes |

Define the send pattern and, if necessary, activate the maximum value (this value is not retained after a reset).

| | |
|---|---|
| Send pattern | • never<br>• periodically<br>• on change<br>• on change and periodically |
| on change of<br>*(if sent on change)* | 2% • 5% • 10% • 25% • 50% |
| Send cycle<br>*(if sent periodically)* | 5 s ... 2 h; 10 s |
| Use maximum value | No • Yes |

### Beaufort scale

| Beaufort | Meaning |
|---|---|
| 0 | Calm |
| 1 | Light air |
| 2 | Light breeze |
| 3 | Gentle breeze |
| 4 | Moderate breeze |
| 5 | Fresh breeze |
| 6 | Strong breeze |
| 7 | High wind |
| 8 | Gale |
| 9 | Severe gale |
| 10 | Storm |
| 11 | Violent storm |
| 12 | Hurricane |

# 6.3. Wind threshold values

Activate the wind threshold values required (maximum four) The menus for the further setting of the threshold values are then displayed.

| | |
|---|---|
| Threshold value 1 | No • Yes |
| Threshold value... | No • Yes |
| Threshold value 4 | No • Yes |

## 6.3.1. Wind threshold value 1-4

### Threshold value

Set, in which cases threshold values and delay times received are to be kept per object. The parameter is only taken into consideration if the specification/ setting by object is activated further down. Please note that the setting "After power restoration and programming" should not be used for the initial start-up, as the factory settings are always used until the first call (setting via objects is ignored).

| | |
|---|---|
| Maintain the | |
| threshold values and delays received via communication objects | • never<br>• after power supply restoration<br>• after power supply restoration and programming |
| . | |

Select whether the threshold value is to be specified per parameter or via a communication object.

| | |
|---|---|
| Threshold value setpoint using | Parameter • Communications object |

When the **threshold value per parameter** is specified, then the value is set.

| | |
|---|---|
| Threshold value in 0.1 m/s | 1 ... 350; 40 |

When the **threshold value per communication object** is specified, the starting value, object value limit and type of change to the threshold value are then set.

| | |
|---|---|
| Start threshold value in 0.1 m/s<br>valid until first call | 1 ... 350; 40 |
| Object value limit (min.) in 0.1 m/s increments | 1 ... 350 |
| Object value limit (max.) in 0.1 m/s increments | 1 ... 350 |
| Type of threshold change | Absolute value • Increase/decrease |
| Step size<br>*(upon increase/decrease change)* | 0.1 m/s • 0.2 m/s • 0.5 m/s • 1.0 m/s • 2.0 m/s • 5.0 m/s |

With both of the methods for specifying the threshold values the hysteresis is set.

| | |
|---|---|
| Hysteresis setting | in % • absolute |
| Hysteresis in % (relative to threshold value)<br>*(for setting in %)* | 0 ... 100; 50 |
| Hysteresis in 0.1 m/s<br>*(for absolute setting)* | 0 ... 350; 20 |

## Switching output

Define which value the output transmits if the threshold value is exceeded or undercut. Set the delay for the switching and in which cases the switch output transmits.

| | |
|---|---|
| When the following conditions apply, the output is<br>(LV = Threshold value) | • GW above = 1 \| GW - Hyst. below = 0<br>• GW above = 0 \| GW - Hyst. below = 1<br>• GW below = 1 \| GW + Hyst. above = 0<br>• GW below = 0 \| GW + Hyst. above = 1 |
| Delays can be set via objects<br>(in seconds) | No • Yes |
| Delay from 0 to 1 | none • 1 s ... 2 h |
| Delay from 1 to 0 | none • 1 s ... 2 h |
| Switching output sends | • on change<br>• on change to 1<br>• on change to 0<br>• on change and periodically<br>• on change to 1 and periodically<br>• on change to 0 and periodically |
| Cycle<br>*(if sent periodically)* | 5 s ... 2 h |

## Block

If necessary, activate the switching output block and set what a 1 or 0 at the block entry means and what happens in the event of a block.

| | |
|---|---|
| Use switching output block | No • Yes |
| Analysis of the blocking object | • At value 1: block \| At value 0: release<br>• At value 0: block \| At value 1: release |
| Blocking object value before first call | 0 • 1 |
| Action when locking | • Do not send message<br>• send 0<br>• send 1 |
| Action upon release<br>(with 2 seconds release delay) | [Dependent on the "Switching output sends" setting] |

The behaviour of the switching output on release is dependent on the value of the parameter "Switching output sends" (see "Switching output")

| | |
|---|---|
| Switching output sends on change | do not send message •<br>Status object/s send/s |

| | |
|---|---|
| Switching output sends on change to 1 | do not send message •<br>If switching output = 1 ➔ send 1 |
| Switching output sends on change to 0 | do not send message • |
| | If switching output = 0 ➔ send 0 |
| Switching output sends on change and periodically | Send switching output status |
| Switching output sends on change to 1 and periodically | If switching output = 1 ➔send 1 |
| Switching output sends on change to 0 and periodically | If switching output = 0 ➔send 0 |

## 6.4. Computer

Activate the multi-functional computer, with which the input data can be changed by calculation, querying a condition or converting the data point type. The menus for the further setting of the computer are then displayed.

| | |
|---|---|
| Computer 1 | No • Yes |
| Computer... | No • Yes |
| Computer 8 | No • Yes |

### 6.4.1. Computers 1-8

Set, in which cases input values received are to be kept per object. Please note that the setting "After power restoration and programming" should not be used for the initial start-up, as the factory settings are always used until the first call (setting via objects is ignored).

| | |
|---|---|
| Maintain the | |
| input values received via communication objects | • never<br>• after power supply restoration<br>• after power supply restoration and programming |
| | |

Select the function set the input mode and starting values for input 1 and input 2.

| | |
|---|---|
| Function (I = Input) | • Prerequisite: E1 = E2<br>• Prerequisite: E1 > E2<br>• Prerequisite: E1 >= E2<br>• Prerequisite: E1 < E2<br>• Prerequisite: E1 <= E2<br>• Prerequisite: E1 - E2 >= E3<br>• Prerequisite: E2 - E1 >= E3<br>• Prerequisite: E1 - E2 amount >= E3<br>• Calculation: E1 + E2<br>• Calculation: E1 - E2<br>• Calculation: E2 - E1<br>• Calculation: E1 - E2 Amount<br>• Calculation: Output 1 = E1 × X + Y \| Output 2 = E2 × X + Y \|<br>• Transformation: General |
| Tolerance for comparison<br>*(in the case of prerequisite E1 = E2)* | 0 ... 4,294,967,295 |
| Input type | [Selection options depending on the function]<br>• 1 bit<br>• 1 byte (0...255)<br>• 1 byte (0%...100%)<br>• 1 byte (0°...360°)<br>• 2 byte counter without math. symbol<br>• 2 byte counter with math. symbol<br>• 2 byte floating point<br>• 4 byte counter without math. symbol<br>• 4 byte counter with math. symbol<br>• 4 byte floating point |
| Starting value E1 / E2 / E3 | [Input range depending on the type of input] |

### *Prerequisites*

When querying the prerequisites set the output type and output values at different statuses:

| | |
|---|---|
| Output type | • 1 bit<br>• 1 byte (0...255)<br>• 1 byte (0%...100%)<br>• 1 byte (0°...360°)<br>• 2 byte counter without math. symbol<br>• 2 byte counter with math. symbol<br>• 2 byte floating point<br>• 4 byte counter without math. symbol<br>• 4 byte counter with math. symbol<br>• 4 byte floating point |
| Output value *(if applicable output value A1 / A2)* | |

| | |
|---|---|
| if the condition is met | 0 [Input range depending on the type of output] |
| if the condition is not met | 0 [Input range depending on the type of output] |
| if the monitoring time period is exceeded | 0 [Input range depending on the type of output] |
| if blocked | 0 [Input range depending on the type of output] |

Set the output send pattern.

| | |
|---|---|
| Output sends | • on change<br>• on change and after reset<br>• on change and periodically<br>• when receiving an input object<br>• when receiving an input object and periodically |
| Type of change<br>*(is only sent if "on change" is selected)* | • on each change<br>• on change to condition met<br>• on change to condition not met |
| Send cycle<br>*(if sent periodically)* | 5 s ... 2 h; 10 s |

Set the text to be displayed for conditions met / not met.

| | |
|---|---|
| Text if the condition is met | [Free text max. 14 chars.] |
| Text if the condition is not met | [Free text max. 14 chars.] |

If applicable set the send delays.

| | |
|---|---|
| Send delay in the event of change to the condition is met | none • 1 s • ... • 2 h |
| Send delay in the event of change to the condition is not met | none • 1 s • ... • 2 h |

### *Calculations and transformation*

For calculations and transformations set the output values to the various conditions:

| Output value (if applicable A1 / A2) | |
|---|---|
| if the monitoring time period is exceeded | 0 [Input range depending on the type of output] |
| if blocked | 0 [Input range depending on the type of output] |

Set the output send pattern.

| | |
|---|---|
| Output sends | • on change<br>• on change and after reset<br>• on change and periodically<br>• when receiving an input object<br>• when receiving an input object and periodically |
| on change of<br>*(only if calculations are transmitted for changes)* | 1 ... [Input range depending on the type of input] |
| Send cycle<br>*(if sent periodically)* | 5 s ... 2 h; 10 s |

For **Calculations of the form output 1 = E1 × X + Y | output 2 = E2 × X +Y** define the variables X and Y. The variables can have a positive or negative sign, 9 digits before and 9 digits after the decimal point.

| | |
|---|---|
| Formula for output A1: A1 = E1 × X + Y | |
| X | 1.00 [free input] |
| Y | 0.00 [free input] |
| Formula for output A2: A2 = E2 × X + Y | |
| X | 1.00 [free input] |
| Y | 0.00 [free input] |

***Further settings for all formulas***

If necessary, activate the input monitoring. Set which inputs are to be monitored, at which intervals the inputs are to be monitored and what value the "monitoring status" should have, if the monitoring period is exceeded without feedback.

| | |
|---|---|
| Use input monitoring | No • Yes |
| Monitoring of | • E1<br>• E2<br>• E3<br>• E1 and E2<br>• E1 and E3<br>• E2 and E3<br>• E1 and E2 and E3<br>[depending on the function] |
| Monitoring period | 5 s • ... • 2 h; 1 min |
| Value of the object "monitoring status" if period is exceeded | 0 • 1 |

If necessary, activate the computer block and set what a 1 or 0 at the block entry means and what happens in the event of a block.

| | |
|---|---|
| Use block | No • Yes |
| Analysis of the blocking object | • At value 1: block \| At value 0: release<br>• At value 0: block \| At value 1: release |

| | |
|---|---|
| Value before first call | 0 • 1 |
| Output pattern<br>On block | • do not send anything<br>• send value |
| On release | • as send pattern [see above]<br>• send current value immediately |

# 6.5. Logic

The device has 16 logic inputs, eight AND and eight OR logic gates.

Activate the logic inputs and assign object values up to first call.

| | |
|---|---|
| Use logic inputs | Yes • No |
| Object value prior to first call for: | |
| - Logic input 1 | 0 • 1 |
| - Logic input ... | 0 • 1 |
| - Logic input 16 | 0 • 1 |

Activate the required logic outputs.

### AND logic

| | |
|---|---|
| AND logic 1 | not active • active |
| AND logic ... | not active • active |
| AND logic 8 | not active • active |

### OR logic

| | |
|---|---|
| OR logic 1 | not active • active |
| OR logic ... | not active • active |
| OR logic 8 | not active • active |

## 6.5.1. AND logic 1-8 and OR logic outputs 1-8

The same setting options are available for AND and OR logic.

Each logic output may transmit one 1 bit or two 8 bit objects. Determine what the out put should send if logic = 1 and = 0.

| | |
|---|---|
| 1. / 2. / 3. / 4. Input | • do not use<br>- Logic inputs 1...16<br>- Logic inputs 1...16 inverted<br>• all switching events that the device provides (see *Connection inputs of the AND/OR logic*) |
| Output type | • a 1-Bit-object<br>• two 8-bit objects |

If the **output type is a 1-bit object**, set the output values for the various conditions.

| | |
|---|---|
| Output value<br>if logic = 1 | 1 • 0 |
| Output value<br>if logic = 0 | 1 • 0 |
| Output value<br>If block is active | 1 • 0 |
| Output value if<br>monitoring period is exceeded | 1 • 0 |

If the **output type is two 8-bit objects**, set the type of object and the output values for the various conditions.

| | |
|---|---|
| Object type | • Value (0…255)<br>• Percent (0…100%)<br>• Angle (0…360°)<br>• Scene call-up (0…127) |
| Output value object A<br>if logic = 1 | 0 … 255 / 100% / 360° / 127; 1 |
| Output value object B<br>if logic = 1 | 0 … 255 / 100% / 360° / 127; 1 |
| Output value object A<br>if logic = 0 | 0 … 255 / 100% / 360° / 127; 0 |
| Output value object B<br>if logic = 0 | 0 … 255 / 100% / 360° / 127; 0 |
| Output value object A<br>if block is active | 0 … 255 / 100% / 360° / 127; 0 |
| Output value object B<br>if block is active | 0 … 255 / 100% / 360° / 127; 0 |
| Output value object A if<br>monitoring period is exceeded | 0 … 255 / 100% / 360° / 127; 0 |
| Output value object B if<br>monitoring period is exceeded | 0 … 255 / 100% / 360° / 127; 0 |

Set the output send pattern.

| | |
|---|---|
| Send pattern | • on change of logic<br>• on change of logic to 1<br>• on change of logic to 0<br>• on change of logic and periodically<br>• on change of logic to 1 and periodically<br>• on change of logic to 0 and periodically<br>• on change of logic+object receipt<br>• on change of logic+object receipt and periodically |
| Send cycle<br>*(if sent periodically)* | 5 s • 10 s • ... • 2 h |

### Block

If necessary, activate the block for the logic output and set what a 1 or 0 at the block input means and what happens in the event of a block.

| | |
|---|---|
| Use block | No • Yes |
| Analysis of the blocking object | • At value 1: block \| At value 0: release<br>• At value 0: block \| At value 1: release |
| Blocking object value before first call | 0 • 1 |
| Output pattern<br>On block | • Do not send message<br>• Transmit block value [see above, Output value if blocking active] |
| On release<br>(with 2 seconds release delay) | [send value for current logic status] |

### Monitoring

If necessary, activate the input monitoring. Set which inputs are to be monitored, at which intervals the inputs are to be monitored and what value the "monitoring status" should have, if the monitoring period is exceeded without a feedback being given.

| | |
|---|---|
| Use input monitoring | No • Yes |
| Input monitoring | • 1 • 2 • 3 • 4<br>• 1 + 2 • 1 + 3 • 1 + 4 • 2 + 3 • 2 + 4 • 3 + 4<br>• 1 + 2 + 3 • 1 + 2 + 4 • 1 + 3 + 4 • 2 + 3 + 4<br>• 1 + 2 + 3 + 4 |
| Monitoring period | 5 s • ... • 2 h; 1 min |
| Output behaviour on exceeding the monitoring time | • Do not send message<br>• Send value exceeding [= value of the parameter "monitoring period"] |

## 6.5.2. AND logic connection inputs

Do not use
Logic input 1
Logic input 1 inverted
Logic input 2
Logic input 2 inverted
Logic input 3
Logic input 3 inverted
Logic input 4
Logic input 4 inverted
Logic input 5
Logic input 5 inverted
Logic input 6
Logic input 6 inverted
Logic input 7
Logic input 7 inverted
Logic input 8

Logic input 8 inverted
Logic input 9
Logic input 9 inverted
Logic input 10
Logic input 10 inverted
Logic input 11
Logic input 11 inverted
Logic input 12
Logic input 12 inverted
Logic input 13
Logic input 13 inverted
Logic input 14
Logic input 14 inverted
Logic input 15
Logic input 15 inverted
Logic input 16
Logic input 16 inverted
Wind Sensor malfunction ON
Wind sensor malfunction OFF
Wind switching output 1
Wind switching output 1 inverted
Wind switching output 2
Wind switching output 2 inverted
Wind switching output 3
Wind switching output 3 inverted
Wind switching output 4
Wind switching output 4 inverted

### 6.5.3. Connection inputs of the OR logic

The OR logic connection inputs correspond to those of the AND logic. In addition, the following inputs are available for the OR logic:

AND logic output 1
AND logic output 1 inverted
AND logic output 2
AND logic output 2 inverted
AND logic output 3
AND logic output 3 inverted
AND logic output 4
AND logic output 4 inverted
AND logic output 5
AND logic output 5 inverted
AND logic output 6
AND logic output 6 inverted
AND logic output 7
AND logic output 7 inverted
AND logic output 8
AND logic output 8 inverted

**Elsner Elektronik GmbH** Control and Automation Engineering
Sohlengrund 16
75395 Ostelsheim
Germany
Phone +49 (0) 70 33 / 30 945-0 info@elsner-elektronik.de
Fax +49 (0) 70 33 / 30 945-20 www.elsner-elektronik.de