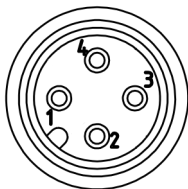


FEUCHTE-, TEMPERATURSENSOR

RS485 · MODBUS RTU · QUICK START

01 ELEKTRISCHER ANSCHLUSS / Electrical connection


PIN	Signal	Funktion	Anschluss
1	+ UB	4 ... 18 V DC	Versorgung +
2	RS485 B / Data-	RS485 B	B / D-
3	GND	0 V / Masse	GND
4	RS485 A / Data+	RS485 A	A / D+
WARNUNG Versorgung und RS485-Leitungen vor dem Einschalten prüfen. Eine falsche Verdrahtung kann das Gerät oder den RS485-Adapter beschädigen.			

Sensorkabel diverser Hersteller in M12 x1mm Buchse, A-Kodierung verwenden.

02 MONTAGE UND HANDHABUNG / Installation

- Sensorkopf frei anströmen lassen und Temperaturgradienten entlang des Gehäuses vermeiden.
- Filtereinsatz im Betrieb nicht abdecken oder mechanisch belasten.
- RS485 als Bus verdrahten. Bei langen Leitungen Abschluss nur an den beiden Busenden vorsehen.

03 KOMMUNIKATION UND ERSTER TEST / Modbus setup

Baudrate	Datenbits	Parität	Stopbits	Slave-ID
9600	8	None	1	234

Empfohlener Funktionstest

Funktion	Start	Anzahl	Datentyp	Auswertung
FC03	4000	4	INT16	siehe Seite 2

HINWEIS Vollständige Kommunikationseinstellungen, Änderung der Slave-ID sowie alle Register sind in der „Modbus Programmierhilfe BRH 100DC“ beschrieben.

04 KOMPAKTE REGISTERÜBERSICHT / Holding registers · FC03/FC04

Messwert	INT16	Skalierung	FLOAT32	Einheit
Temperatur	4000	÷100	1000/1001	°C
Relative Feuchte	4001	÷100	1002/1003	%RH
Taupunkt	4002	÷100	1004/1005	°C
Abs. Feuchte	4003	÷100	1006/1007	g/m ³
Temperatur Fahrenheit	4014	÷100	1026/1027	°F
Taupunkt Fahrenheit	4015	÷100	1028/1029	°F
Mischungsverhältnis	4016	÷100	1030/1031	g/kg

FLOAT32: zwei Register pro Wert, Wortreihenfolge Low-Word → High-Word.

05 KURZE CODEBEISPIELE / Python · pymodbus

INT16 lesen

```
from pymodbus.client import ModbusSerialClient
c = ModbusSerialClient(port='COM5', baudrate=9600,
    parity='N', stopbits=1, bytesize=8, timeout=.5)
if c.connect():
    r = c.read_holding_registers(4000, 4, slave=235)
    t = r.registers[0] / 100.0
    rh = r.registers[1] / 100.0
    print(t, rh)
c.close()
```

FLOAT32 lesen

```
import struct

def regs_to_float(lo, hi):
    raw = struct.pack('<HH', lo & 0xFFFF, hi & 0xFFFF)
    return struct.unpack('<F', raw)[0]
r = c.read_holding_registers(1000, 4, slave=235)
t = regs_to_float(r.registers[0], r.registers[1])
rh = regs_to_float(r.registers[2], r.registers[3])
```

Installation: pip install pymodbus

06 LED-STATUS UND SCHNELLE FEHLERPRÜFUNG / Status / Troubleshooting

Anzeige / Beobachtung	Bedeutung / Prüfung
Grün langsam blinkend	Normalbetrieb, Messwerte gültig, Modbus bereit.
Blau kurzer Puls	Modbus-Telegramm empfangen bzw. beantwortet.
Gelb blinkend	Messwertwarnung; u. a. relative Feuchte > 80 %RH.
Rot langsam blinkend	HDC3020 nicht erreichbar oder dauerhafter Sensor-/CRC-Fehler.
Blau / Rot wechselnd	Bus-/Kommunikationsfehler; Verdrahtung, Abschluss und Telegrammqualität prüfen.
Keine Antwort	COM-Port, Versorgung, Slave-ID 235, 9600 8N1 sowie A/B prüfen.