

UMB-ASCII 2.0
Universal-Measurement-Bus
Kommunikationsprotokoll
für meteorologische Sensoren

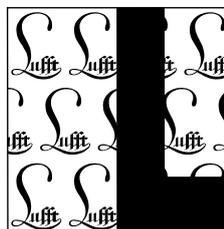
WS3000-UMB
WS3100-UMB
WS100-UMB
VS100k-UMB
SHM31-UMB



Stand 26.02.2018
Protokoll-Version 2.0
Dokument-Version 1.1

G. LUFFT MESS- UND
REGELTECHNIK GMBH

POSTFACH 4252
70719 FELLBACH
TEL. 49 (711) - 51822-0
FAX 49 (711) - 51822-41



Lufft

Inhaltsverzeichnis

| | | |
|--------|---|----|
| 1 | Versionsgeschichte..... | 3 |
| 2 | Vorbemerkung..... | 4 |
| 3 | Syntax | 5 |
| 3.1 | Prüfsumme | 5 |
| 3.2 | Beispiele | 6 |
| 4 | Befehle..... | 7 |
| 4.1 | Übersicht | 7 |
| 4.2 | Fehlercodes | 7 |
| 4.3 | Kommandos..... | 9 |
| 4.3.1 | Auto Transmit Intervall..... | 9 |
| 4.3.2 | Auto Transmit Mode | 9 |
| 4.3.3 | Baudrate | 9 |
| 4.3.4 | Gerätebeschreibung..... | 9 |
| 4.3.5 | Geräte-Id..... | 10 |
| 4.3.6 | Gerätename..... | 10 |
| 4.3.7 | Blocktrennzeichen | 10 |
| 4.3.8 | Zeilenendezeichen..... | 10 |
| 4.3.9 | Dezimaltrennzeichen..... | 10 |
| 4.3.10 | Telegramm-Ende-Zeichen | 11 |
| 4.3.11 | Parametertrennzeichen..... | 11 |
| 4.3.12 | Telegramm-Start-Zeichen..... | 11 |
| 4.3.13 | Protokoll | 11 |
| 4.3.14 | Parität | 12 |
| 4.3.15 | Seriennummer | 12 |
| 4.3.16 | EEPROM..... | 12 |
| 4.3.17 | Reset | 12 |
| 4.3.18 | Reset auf Werkseinstellungen..... | 14 |
| 4.3.19 | Geräteinformationen..... | 14 |
| 4.3.20 | Temporär auf UMB-Binär umschalten | 14 |
| 4.3.21 | Kanalabfrage..... | 15 |
| 4.3.22 | Standard-Set Abfrage | 15 |
| 4.3.23 | Tunnel..... | 15 |

1 Versionsgeschichte

| Dokument-Version | Datum | Bearbeiter | Änderungsbeschreibung |
|------------------|------------|------------|--|
| 0.0 | 11.07.2016 | RZF | Erstellung |
| 0.1 | 05.09.2016 | ZI | einschränkender Hinweis zu den Kommandobeschreibungen PDS, PBS & PPS |
| 0.2 | 24.01.2017 | RZF | Tunnel-Befehl hinzugefügt |
| 1.0 | 07.03.2017 | RZF | Pin für EEPROM Befehle hinzugefügt |
| 1.1 | 14.02.2018 | GS/ITSC | Formulierungen StandardSet Befehl und Telegrammlängen geändert |
| | 26.02.2018 | RZF | Fehlercodes ergänzt |

2 Vorbemerkung

UMB-ASCII 2.0 ist ein menschenlesbares Protokoll zur Abfrage von meteorologischen Sensoren. Bei der Definition des Protokolls wurde größtmöglicher Wert auf Anpassbarkeit gelegt, um auf die Anforderungen verschiedenster Messwerterfassungsanlagen eingehen zu können.

Das Protokoll erlaubt einen automatischen Telegrammversand, so dass eine Messwerterfassung auch ohne vorherige Anfrage möglich ist.

3 Syntax

Teile in eckigen Klammern sind optional:

Anfrage <Add>:<Nr>:<Payload><CR><LF>

Antwort <STX><Add>:<Nr>:<Payload>:<Status>:<Checksum><CR><LF><EOT>

Payload <Cmd>[;<Param₀>;...;Param_N][=<Value₀>;...;<Value_N>]

Die einzelnen Blöcke des Protokolls sind durch einen Doppelpunkt getrennt. Dieses Blocktrennzeichen ist einstellbar. <Add> ist die UMB-Adresse des Sensors, vier Hex-Zeichen, mit führenden Nullen, im Bereich 0001-FFFF. Nachrichten mit falscher Adresse werden vom Sensor ignoriert. Das Feld <Nr> kann bei der Anfrage vom Sender auf zwei beliebige Hex-Zeichen gesetzt werden. Der Sensor übernimmt diese Zeichen und fügt sie in seine Antwort ein. <Nr> kann z.B. als zeitliche Referenz genutzt werden, wenn die Möglichkeit besteht, dass Anfrage und Antwort nicht direkt hintereinander ankommen. <Payload> sind die eigentlichen Nutzdaten des Telegramms. In ihr steht der Befehl mit seinen Parametern und Werten. Das <Status>-Feld besteht aus zwei Hex-Zeichen, mit führenden Nullen. Bei unbekanntem oder fehlerhaftem Befehl enthält die Payload den Befehl und Status den entsprechenden Fehlercode. Anfragen und Antworten enden, einstellbar, mit den Zeichen <CR><LF>. Die Antwort eines Sensors ist noch zusätzlich von den Steuerzeichen <STX> und <EOT> eingerahmt.

Die Payload enthält den Befehl mit seinen Informationen. Befehle können einen oder mehrere Parameter haben, durch ein Semikolon getrennt. Dieses Parametertrennzeichen ist einstellbar. Enthält die Payload einer Anfrage ein Gleichheitszeichen, so ist es ein Schreibbefehl. Der Wert <Value> kann, je nach Befehl, einer der unterstützten Datentypen, incl. String sein. Auch hier können mehrere Werte durch Semikolon getrennt übermittelt werden. Als Dezimaltrennzeichen wird, einstellbar, ein Punkt verwendet.

Die Länge einer Anfrage ist auf 128 Zeichen, die Länge einer Antwort 512 Zeichen begrenzt. Geräteabhängig können größere Längen zugelassen werden. Solche Ausnahmen sind im jeweiligen Gerätehandbuch definiert.

3.1 Prüfsumme

Antworten vom Sensor haben eine Prüfsumme <Checksum>, um die Integrität der Daten sicherzustellen. Die Prüfsumme ist das Zweierkomplement der 8-Bit-Summe aller Zeichen, incl. Steuerzeichen <STX> und <EOT>, ohne die Prüfsumme selbst. Sie wird durch zwei Hex-Zeichen mit führender Null dargestellt.

3.2 Beispiele

Für die folgenden Beispiele wird angenommen: <Add> = 7001, <Nr> =4E

Reset 7001:4E:RST<CR><LF>
 <STX>7001:4E:RST:00:<Checksum><CR><LF><EOT>

Read Float 7001:4E:EF;120<CR><LF>
 <STX>7001:4E:EF;120=34.5654:00:<Checksum><CR><LF><EOT>

WriteByte 7001:4E:ES;34=1<CR><LF>
 <STX>7001:4E:ES;34=1:00:<Checksum><CR><LF><EOT>

Read Channel 7001:4E:CHN;100<CR><LF>
 <STX>7001:4E:CHN;100=+23.45:00:<Checksum><CR><LF><EOT>

 7001:4E:CHN;110<CR><LF>
 <STX>7001:4E:CHN;110:28:<Checksum><CR><LF><EOT> (Channel busy)

 7001:4E:CHN;523<CR><LF>
 <STX>7001:4E:CHN;523:24:<Checksum><CR><LF><EOT> (Channel unknown)

4 Befehle

4.1 Übersicht

Die Parameter des einheitlichen Teils der UMB-E2-Liste (1-199) sind durch entsprechende Befehle einstellbar. Jedes Gerät kann zusätzliche, eigene Befehle spezifizieren.

EERPOM-Befehle sind nur zur internen Verwendung vorhanden und werden nicht an Kunden kommuniziert (siehe UMB-Monitor).

| Befehl | | R/W | Beschreibung |
|---------------------------|------------------------------|-----|---|
| Allgemeine Befehle | | | |
| ATI | | RW | Auto Transmit Intervall |
| ATM | | RW | Auto Transmit Mode |
| BAU | BAUdrate | RW | Baudrate |
| DSC | Device DeSCription | RW | Geräte-Beschreibung |
| ID | | RW | Geräte-ID |
| NAM | Device NAME | R | Geräte-Name |
| PBS | Protocol Block Separator | RW | Blocktrennzeichen (:) |
| PCR | Protocol Carriage Return | RW | Zeilenende-Zeichen(CRLF) |
| PDS | Protocol Decimal Separator | RW | Dezimaltrennzeichen (.) |
| PEN | Protocol ENd | RW | Ende-Zeichen (EOT) |
| PPS | Protocol Parameter Separator | RW | Parametertrennzeichen (;) |
| PRT | PRoTocol | RW | Protokoll |
| PRY | PaRitY | RW | Paritätseinstellung |
| PST | Protocol STart | RW | Start-Zeichen (STX) |
| SRN | SeRialNumber | R | Luft Seriennummer |
| | | | |
| ES;<Add>;<Pin> | | RW | EEPROM uint8 |
| EI;<Add>;<Pin> | | RW | EEPROM uint16 |
| EL;<Add>;<Pin> | | RW | EEPROM uint32 |
| EF;<Add>;<Pin> | | RW | EEPROM float32 |
| ED;<Add>;<Pin> | | RW | EEPROM float64 |
| | | | |
| RST[=<Value>] | ReSeT | RW | Reset, sofort / nach <Value> ms |
| RSD | ReSet Default | R | Reset auf Werkseinstellungen |
| IFO;<Info> | InFO | R | Geräteinformationen |
| UMB | | | Temporär auf UMB-Binär umschalten |
| | | | |
| CHN;<ChnNum> | CHaNnel | R | Kanal <ChnNum> lesen |
| SS;<Num> | | R | Standard Set <Num> lesen |
| TNL;<BusNum>;<Len> | TuNnel | W | Daten der Länge <Len> nach <BusNum> tunneln |

4.2 Fehlercodes

Die Werte des <Status>- Feldes entsprechen den die im UMB-Binär-Protokoll spezifizierten Fehlercodes, wobei jedoch nicht alle verwendet werden:

| <status> | Define | Beschreibung |
|----------|---------------|---|
| 00h | OK | Kommando erfolgreich; kein Fehler; alles i.O. |
| | | |
| 10h | UNBEK_CMD | unbekanntes Kommando; wird von diesem Gerät nicht unterstützt |
| 11h | UNGLTG_PARAM | ungültige Parameter |
| 12h | UNGLTG_HEADER | ungültige Header-Version |
| 13h | UNGLTG_VERC | ungültige Version des Befehls |
| 14h | UNGLTG_PW | Passwort für Kommando falsch |
| 15h | UNGLTG_WERT | ungültiger Wert |
| | | |
| 20h | LESE_ERR | Lesefehler |
| 21h | SCHREIB_ERR | Schreibfehler |
| 22h | ZU_LANG | Länge zu groß; max. zulässige Länge wird in <maxlength> angegeben |

| | | |
|-----------|--------------------|--|
| 23h | UNGLTG_ADRESS | ungültige Adresse / Speicherstelle |
| 24h | UNGLTG_KANAL | ungültiger Kanal |
| 25h | UNGLTG_CMD | Kommando in diesem Modus nicht möglich |
| | | |
| 26h | UNBEK_CAL_CMD | unbekanntes Test-/Abgleich-Kommando |
| 27h | CAL_ERROR | Fehler bei der Kalibrierung |
| 28h | BUSY | Gerät nicht bereit; z.B. Initialisierung / Kalibrierung läuft |
| 29h | LOW_VOLTAGE | Unterspannung |
| 2Ah | HW_ERROR | Hardwarefehler |
| 2Bh | MEAS_ERROR | Fehler in der Messung |
| 2Ch | INIT_ERROR | Fehler bei der Geräteinitialisierung |
| 2Dh | RTOS_ERROR | Fehler im Betriebssystem |
| 2Eh | COM_ERROR | Kommunikationsfehler auf einer internen Schnittstelle |
| 2Fh | HW_SW_MISMATCH | Hardware und Software Version passen nicht |
| | | |
| 30h | E2_DEFAULT_KONF | Fehler in der Konfiguration, Default-Konfiguration wurde geladen |
| 31h | E2_CAL_ERROR | Fehler im Abgleich / der Abgleich ist ungültig, Messung nicht möglich |
| 32h | E2_CRC_KONF_ERR | CRC-Fehler beim Laden der Konfiguration; Default-Konfiguration wurde geladen |
| 33h | E2_CRC_KAL_ERR | CRC-Fehler beim Laden der Abgleich-Daten; Messung nicht möglich |
| 34h | ADJ_STEP1 | Abgleich Step 1 |
| 35h | ADJ_OK | Abgleich OK |
| 36h | KANAL_AUS | Kanal deaktiviert |
| 37h | SERVICE_MODE | Service Modus aktiv |
| | | |
| 50h | VALUE_OVERFLOW | Messgröße (+Offset) liegt außerhalb des eingestellten Darstellungsbereichs |
| 51h | VALUE_UNDERFLOW | |
| 52h | CHANNEL_OVERRANGE | Messwert (physikalisch) liegt außerhalb des Messbereichs (z.B. ADC-Ovrange) |
| 53h | CHANNEL_UNDERRANGE | |
| 54h | DATA_ERROR | Datenfehler in den Messdaten oder keine gültigen Daten vorhanden |
| 55h | MEAS_UNABLE | Gerät / Sensor kann auf Grund der Umgebungsbedingungen keine gültige Messung durchführen |
| | | |
| 0x60 | FLASH_CRC_ERR | CRC-Fehler in den Flash-Daten |
| 0x61 | FLASH_WRITE_ERR | Fehler beim Schreiben ins Flash; z.B. Speicherstelle nicht gelöscht |
| 0x62 | FLASH_FLOAT_ERR | Flash enthält ungültige Float-Werte |
| 0x63 | FLASH_ERR | Flash defekt, Initialisierungsfehler |
| 0x64 | CONF_ERR | Fehlerhafte Konfiguration |
| F0h - FEh | Reserviert!! | reservierter Bereich für z.B. LCOM |
| FFh | UNBEK_ERR | unbekannter Fehler |

4.3 Kommandos

Die Kommandos sind case-insensitive, d.h. es wird nicht zwischen Groß- und Kleinschreibung unterschieden: RST = rst = rSt.

4.3.1 Auto Transmit Intervall

Kommando <cmd>: ATI

Parameter: keine

Aufruf: ATI ATI=<time>

Antwort: ATI=<time>

Beschreibung: dieses Kommando liest / setzt den Abstand [sek] zwischen zwei automatisch versendeten Standard-Set-Telegrammen.

4.3.2 Auto Transmit Mode

Kommando <cmd>: ATM

Parameter: keine

Aufruf: ATM ATM=<setNumber>

Antwort: ATM=<setNumber>

Beschreibung: dieses Kommando liest / setzt die Nummer des Standard-Sets das automatisch versendet werden soll. Ist <setNumber> 0, so ist der automatische Telegrammversand deaktiviert.

4.3.3 Baudrate

Kommando <cmd>: BAU

Parameter: keine

Aufruf: BAU BAU=<value>

Antwort: BAU=<value>

Beschreibung: dieses Kommando liest / setzt die Baudrate anhand folgender Tabelle:

| <value> | Baudrate |
|---------|----------|
| 2 | 57600 Bd |
| 4 | 28800 Bd |
| 6 | 19200 Bd |
| 8 | 14400 Bd |
| 12 | 9600 Bd |
| 24 | 4800 Bd |
| 48 | 2400 Bd |
| 96 | 1200 Bd |

4.3.4 Gerätebeschreibung

Kommando <cmd>: DSC

Parameter: keine

Aufruf: DSC DSC=<desc>

Antwort: DSC=<desc>⁴⁰

Beschreibung: dieses Kommando liest / schreibt die Gerätebeschreibung mit max. 40 Zeichen. Übrige Zeichen werden mit Leerzeichen aufgefüllt.

4.3.5 Geräte-Id

Kommando <cmd>: ID
Parameter: keine
Aufruf: ID ID=<id>
Antwort: ID=<id>

Beschreibung: dieses Kommando liest / setzt die Geräte-Id im Bereich: 1-255

4.3.6 Gerätename

Kommando <cmd>: NAM
Parameter: keine
Aufruf: NAM
Antwort: NAM=<name>⁴⁰

Beschreibung: dieses Kommando liest den Gerätenamen mit max. 40 Zeichen. Übrige Zeichen werden mit Leerzeichen aufgefüllt.

4.3.7 Blocktrennzeichen

Kommando <cmd>: PBS
Parameter: keine
Aufruf: PBS PBS=<asciiValue>
Antwort: PBS=<asciiValue>

Beschreibung: dieses Kommando liest / setzt den ASCII-Wert des Protokoll-Blocktrennzeichens. Standardwert ist 58, ein Doppelpunkt ':'. Für den ASCII-Wert sind nur Sonderzeichen erlaubt.

4.3.8 Zeilenendezeichen

Kommando <cmd>: PCR
Parameter: keine
Aufruf: PCR PCR=<value>
Antwort: PCR=<value>

Beschreibung: dieses Kommando liest / setzt das Zeilenende-Zeichen anhand folgender Tabelle:

| <value> | Zeilenende |
|---------|-----------------|
| 0 | CRLF (0Dh, 0Ah) |
| 1 | CR (0Ah) |
| 2 | LF (0Dh) |

4.3.9 Dezimaltrennzeichen

Kommando <cmd>: PDS
Parameter: keine
Aufruf: PDS PDS=<asciiValue>
Antwort: PDS=<asciiValue>

Beschreibung: dieses Kommando liest / setzt den ASCII-Wert des Dezimaltrennzeichens. Standardwert ist 46, ein Punkt '.'. Für den ASCII-Wert sind nur Sonderzeichen erlaubt.

4.3.10 Telegramm-Ende-Zeichen

Kommando <cmd>: PEN
Parameter: keine
Aufruf: PEN PEN=<asciiValue>
Antwort: PEN=<asciiValue>

Beschreibung: dieses Kommando liest / setzt den ASCII-Wert des Protokoll-Ende-Zeichens. Standardwert ist 4, ein „end of text“ ‘EOT‘.

4.3.11 Parametertrennzeichen

Kommando <cmd>: PPS
Parameter: keine
Aufruf: PPS PPS=<asciiValue>
Antwort: PPS=<asciiValue>

Beschreibung: dieses Kommando liest / setzt den ASCII-Wert des Protokoll-Parametertrennzeichens. Standardwert ist 59, ein Semikolon ‘;’. Für den ASCII-Wert sind nur Sonderzeichen erlaubt.

4.3.12 Telegramm-Start-Zeichen

Kommando <cmd>: PST
Parameter: keine
Aufruf: PST PST=<asciiValue>
Antwort: PST=<asciiValue>

Beschreibung: dieses Kommando liest / setzt den ASCII-Wert des Protokoll-Start-Zeichens. Standardwert ist 2, ein „start of text“ ‘STX‘.

4.3.13 Protokoll

Kommando <cmd>: PRT
Parameter: keine
Aufruf: PRT PRT=<value>
Antwort: PRT=<value>

Beschreibung: dieses Kommando liest / setzt das Kommunikationsprotokoll anhand folgender Tabelle:

| <value> | Protokoll |
|---------|---------------|
| 0 | UMB-Binär |
| 1 | UMB-ASCII |
| 2 | Terminal |
| 3 | SDI-12 |
| 5 | MODBUS-RTU |
| 6 | MODBUS-ASCII |
| 7 | XDR |
| 9 | UMB-ASCII 2.0 |

Hinweis: Nicht alle Sensoren unterstützen alle genannten Protokolle. Nähere Informationen sind der Beschreibung der Sensoren zu entnehmen.

Achtung: Der Wechsel des Kommunikationsprotokolls wird erst nach einem Neustart wirksam. Danach kann der Sensor nur noch im neuen Protokoll angesprochen werden!

4.3.14 Parität

Kommando <cmd>: PRY
Parameter: keine
Aufruf: PRY PRY=<value>
Antwort: PRY=<value>

Beschreibung: dieses Kommando liest / setzt die Paritätseinstellung anhand folgender Tabelle:

| <value> | Protokoll |
|---------|--------------|
| 0 | keine (8N1) |
| 1 | gerade (8E1) |
| 2 | gerade (7E1) |
| 3 | keine (8N2) |

Hinweis: Die Paritätseinstellung sollte nur für das Modbus und SDI12 verändert werden.

Achtung: Der Wechsel der Paritätseinstellung wird erst nach einem Neustart wirksam!

4.3.15 Seriennummer

Kommando <cmd>: SRN
Parameter: keine
Aufruf: SRN
Antwort: SRN=<serialNumber>

Beschreibung: dieses Kommando gibt die Seriennummer des Geräts in folgendem Format zurück:

<laufende Nummer>³.<Monat Jahr>⁴.<Projektnummer>⁴.<Geräteversion>³

4.3.16 EEPROM

Kommando <cmd>: Ex
Parameter: <address>
Aufruf: Ex;<address> Ex;<address>[;<Pin>]=<value>
Antwort: Ex;<address>=<value>

Beschreibung: dieses Kommando liest / schreibt einen Wert <value> an Adresse <address> des internen EEPROMs. Für sonst schreibgeschützte Bereiche kann ein Pin-Code angegeben werden. 'x' kann dabei für einen der folgenden Datentypen stehen:

| x | Datentyp |
|---|------------------|
| S | Short / uint8 |
| I | Int / uint16 |
| L | Long / uint32 |
| F | Float / float32 |
| D | Double / float64 |

Achtung: Dieser Befehl ist nur für den internen Gebrauch bei Fa. Lufft bestimmt. Unsachgemäße Verwendung kann das Gerät unbrauchbar machen.

4.3.17 Reset

Kommando <cmd>: RST
Parameter: keine
Aufruf: RST RST=<time>
Antwort: RST=<time>

Beschreibung: dieses Kommando löst einen Neustart des Sensors aus. Ist ein Wert in Millisekunden angegeben erfolgt der Neustart nach entsprechender Verzögerung, ansonsten sofort.

4.3.18 Reset auf Werkseinstellungen

Kommando <cmd>: RSD
Parameter: keine
Aufruf: RSD RSD=<time>
Antwort: RSD=<time>

Beschreibung: dieses Kommando löst einen Neustart des Sensors aus. Ist ein Wert in Millisekunden angegeben erfolgt der Neustart nach entsprechender Verzögerung, ansonsten sofort. Alle Benutzereinstellungen werden auf die Werkseinstellungen zurückgesetzt, mit Ausnahme der Geräte-Adresse.

4.3.19 Geräteinformationen

Kommando <cmd>: IFO
Parameter: <info>;<option>
Aufruf: IFO;<info>;<option>
Antwort: IFO;<info>;option>=<value>

Beschreibung: dieses Kommando gibt folgende Geräteinformationen zurück:

| <info> | <option> | Beschreibung | <value> |
|--------|-----------|---------------------------|---|
| 10 | keine | Gerätebezeichnung | <bez> ⁴⁰ z.B. ‚Visibility-Sensor VS20‘ |
| 11 | keine | Gerätebeschreibung | <beschr> ⁴⁰ z.B. ‚Sichtweite A92 West‘ |
| 12 | keine | Hard- und Softwareversion | <hardware> ³ ; <software> ³ Version 2.3 = 17h = 23d |
| 13 | keine | erweiterte Versions-Info | <Lfd.-Nr> ³ ; <MMJJ> ⁴ ; <Projekt> ⁴ ; <Stüli> ³ ; <SPlan> ³ ; <hardware> ³ ; <software> ³ ; <e2version> ³ ; <geräteversion> ³ |
| 14 | keine | Größe des EEPROM | <e2_size> ⁵ |
| 15 | keine | Anzahl verfügbare Kanäle | <channels> ⁵ ; <blocks> ³ max. 75 Kanäle pro Block |
| 16 | <block> | Nummern der Kanäle | <channels> ⁵ ; [<channel> ⁵] ^{<channels>} |
| 20 | <channel> | Messgröße des Kanals | <messgröße> ²⁰ z.B. ‚visibility‘ |
| 21 | <channel> | Messbereich des Kanals | <min> ⁿ ; <max> ⁿ Werte wie Kanal |
| 22 | <channel> | Messeinheit des Kanals | <einheit> ¹⁵ z.B. ‚m‘ |
| 23 | <channel> | Datentyp des Kanals | <date_typ> ² z.B. 16h für float |
| 24 | <channel> | Messwerttyp | <mw_typ> ² z.B. 13h für Mittelwert |
| 30 | <channel> | komplette Kanalinfo | <messgröße> ²⁰ ; <einheit> ¹⁵ ; <mw_typ> ² ; <date_typ> ² ; <min> ⁿ ; <max> ⁿ |

4.3.20 Temporär auf UMB-Binär umschalten

Kommando <cmd>: UMB
Parameter: keine
Aufruf: UMB
Antwort: UMB

Beschreibung: dieses Kommando schaltet das Kommunikationsprotokoll unmittelbar nach Empfang auf UMB-Binär um. Das Protokoll wird nach Ablauf von 10 Minuten automatisch zurück gewechselt.

Achtung: Nach dem Empfang dieses Kommandos kann der Sensor nur noch mit dem UMB-Binär-Protokoll angesprochen werden!

4.3.21 Kanalabfrage

Kommando <cmd>: CHN

Parameter: <channelNumber>

Aufruf: CHN;<channelNumber>

Antwort: CHN;<channelNumber>=<value>

Beschreibung: dieses Kommando liest den Messwert eines bestimmten Kanals. Länge und Format von <value> hängt vom Kanal ab.

4.3.22 Standard-Set Abfrage

Kommando <cmd>: SS

Parameter: <setNumber>

Aufruf: SS;<setNumber>

Antwort: SS;<setNumber>=<value_list>

Beschreibung: dieses Kommando gibt die Messwerte eines Standard-Sets.

Standard-Sets sind abhängig vom Gerätetyp. Ein Gerätetyp kann mehrere Standard-Sets bereitstellen. Die <value_list> eines Standard-Sets kann einen oder mehrere Messwerte, sowie zusätzliche Informationen wie Gerätetyp und Version enthalten. Standard-Sets sind in der jeweiligen Gerätebeschreibung definiert.

4.3.23 Tunnel

Kommando <cmd>: TNL

Parameter: <BusNum>;<Len>

Aufruf: TNL;<BusNum>;<Len>=<Data>^{Len}

Antwort: TNL;<BusNum>;<Len>=<Data>^{Len}

Beschreibung: dieses Kommando tunnelt (binäre) Daten <Data> der Länge <Len> an einen geräte-internen Kommunikationsbus. Gültige Werte für <BusNum> sind in der jeweiligen Gerätebeschreibung definiert. <BusNum> und <Len> sind jeweils zwei Hex-Zeichen.

Achtung: Wird dieses Kommando verwendet, so muss der Parser des Empfängers unbedingt den <Len>-Parameter der Antwort auswerten, da die Bytes von <Data> alle Werte zwischen 0 und 255 annehmen können!

Achtung: <Len> und die Länge von <Data> darf 228 nicht überschreiten!