HART® Field Device Specification

Baumer FlexTop 2222

Revision 0

Document 81311378, rev. 0

Initial release: 1st of March 2019
Current release: 1st of March 2019

® HART is a registered trademark of the HART Communication Foundation

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Scope

Baumer FlexTop 2222 temperature transmitter has built-in support for the HART 7.5 protocol. Since the HART version of the transmitter operates identically with the non-HART version of the transmitter (2212), this document focuses solely on the HART functionalities of the transmitter. For all other operational aspects of the transmitter, please consult the data sheet and the user guide.
This document contains the necessary data for an operator, familiar with the HART protocol, to access all functions of the transmitter from a master system.

1.2 Purpose

This specification is designed to compliment other documentation (e.g., the *FlexTop 2222 User Guide*) by providing a complete, unambiguous description of this Field Device from a HART Communication perspective.

1.3 Who should use this document?

The specification is designed to be a technical reference for HART capable Host Application Developers, System Integrators and knowledgeable End Users. It also provides functional specifications (e.g., commands, enumerations and performance requirements) used during Field Device development, maintenance and testing. This document assumes the reader is familiar with HART Protocol requirements and terminology.

1.4 Abbreviations and definitions

CT	Common Table
DT	Device Specific Table
uint8_t	8-bit unsigned integer, representing value 0 .. 255, can also be used for single bit flags
uint16_t	16-bit unsigned integer, representing value 0 .. 65,535
uint32_t	32-bit unsigned integer, representing value 0 .. 4,294,967,295
float_t	32-bit IEEE-754 (IEC 559) compatible single floating point variable
ASCII	ISO Latin-1 (ISO 8859) string text
packed	HART specific Packed ASCII format
PV	Primary Variable
SV	Secondary Variable
TV	Tertiary Variable
QV	Quaternary Variable
DAC	Digital to Analog Converter
RTD	Resistance Temperature Detector
TC, T/C	Thermocouple

1.5 References

HART Smart Communications Protocol Specification. HCF_SPEC-13. Available from the HCF.
FlexTop 2222, Operating Instruction, FlexTop 22x2, Document 11202169_02. Available from www.baumer.com.

2 DEVICE IDENTIFICATION

Manufacturer Name:	Baumer	**Model Name(s):**	FlexTop 2222
Manufacture ID Code:	96 (60 Hex)	**Device Type Code:**	235 (EB Hex)
HART Protocol Revision	7.5	**Device Revision:**	1
Number of Device Variables	7		
Physical Layers Supported	FSK		
Physical Device Category	Transmitter, Current Output		

The ABC123 is a designed to mount on a DIN-rail. The name plate is located opposite the field terminals and indicates the model name and revision.

3 PRODUCT OVERVIEW

The FlexTop 2212 is a 4…20 mA loop-powered, configurable universal transmitter with galvanic isolation between input and output. The input can be configured for RTD or T/C sensors, resistance, current or voltage signals. Either 2-, 3- or 4-wire connection can be selected for the resistance input. The built-in temperature sensor or an external RTD element can be used to compensate for „cold junction“ (CJC) if thermocouples are connected.The configuration is done with the FlexProgram, and the connection can be established using an USB cable directly mounted between the FlexTop and a PC.The FlexTop 2222 is embedded in silicone which makes it resistant to humid environments. It is ready for direct display mounting through UnitCom cable. Furthermore it has a 6.5 mm center hole for fast sensor replacement and spring loaded mounting screws which ensures a safe fastening even in vibrating environments.
Product highlights:

- HART
- Programmable through integrated USB port
- Sensor calibration for either offset, slope or polynomial adjustment
- Accuracy better than 0.1°C for RTD elements
- Automatic cable compensation calibration (2-wire)
- Fast sampling time < 50 ms
- Galvanic isolated

4 PRODUCT INTERFACES

4.1 Process Interface

4.1.1 Sensor Input Channels

The main temperature sensor ("external sensor") input provides four terminals, marked 1, 2, 3 and 4, for connection of two-, three- or four-wire RTD sensors, or for two wires from a thermocouple. Refer to the Installation Manual for connection details. Operating ranges correspond to the capabilities of each sensor type.
An additional internal temperature sensor is mounted near the sensor terminals. This provides cold junction compensation when a thermocouple is used as the main sensor. Moreover it is possible to mount an external RTD for cold junction compensation.

4.2 Host interface

4.2.1 Analog Output 1: Process Temperature

The two-wire 4-to-20mA current loop is connected on two terminals marked "+" and "-". Refer to the Installation Manual for connection details.
This is the only output from this transmitter, representing the process measurement (temperature, ohm or voltage), linearised and scaled according to the configured range of the instrument. This output corresponds to the Primary Variable. HART Communication is supported on this loop.

A guaranteed linear over-range is provided. Device malfunction can be indicated by down-scale or up-scale current. The direction is selectable by the user; see Section 4.3 below. Current values are shown in the table below.

	Direction	Values (percent of range)	Values (mA or V)
Linear over-range	Down	-3.13% to -0.01%	3.5 to 3.99 mA
	Up	+100.1% to 118.75%	20.01 to 23.00 mA
Device malfunction indication	Fixed value	-3.13% to 118.75%	3.50 to 23.00 mA
Maximum current		+118.75%	23.0 mA
Multi-Drop current draw			4.0 mA
Lift-off voltage			7.0 V

4.3 Local Interfaces

4.3.1 Displays

The FlexTop 2212 has a plug for direct mounting to the DFON touch display from Baumer. The connection is established by using the provided flat ribbon cable provided with the display unit. Please contact Baumer for more information.

Baumer A/S, Runetoften 19, 8210 Aarhus V., Denmark
Tlf +45 8931 7611, fax: +45 8931 7610

4.3.2 Configuration

The FlexTop 2212 can be configured using standard micro USB cable. The maximum distance depends on the cable type, resistance and capacity. A PC and Baumer FlexProgram must be used for this.
To be able to configure the FlexTop 2222 through the micro USB cable the dedicated software, FlexProgram, must be downloaded to the PC. The software is available at www.baumer.com
During the configuration the data are transferred from the PC to the FlexTop 2222, where it is stored in the internal memory.
Error indications during configuration:

1) An error message is showed on the PC.

With the FlexProgram either 2-, 3- or 4-wire connection can be selected for the resistance input. The built-in temperature sensor, an external RTD element or a fixed value can be used to compensate for "cold junction" (CJC) if thermocouples are connected.
With the FlexProgram it is possible to calibrate the RTD or T/C element for either offset (1-point), slope (2-point) or polynomial (3-point) adjustment. Linearization, damping, sampling time and status indication can be configured too. Furthermore it is possible to do an automatic cable compensation calibration when using 2-wire.

5 DEVICE VARIABLES

Seven Device Variables are implemented.

DV No.	Name	Description	Unit codes		Classification code
5, 244	Percent of range	Conductivity measurement with temperature compensation Current Output capability on current output channels: 2 (indirectly via Dev. Var. 4)	57	%	0 Not Class'd
6, 245	Loop current	Loop Current associated with PV, representing sensor measurement	39	mA	84 Current
0, 246	PV	Temperature, resistance or voltage of external sensor (depeding on selected sensor type).	32 33 35 36 37	^{0}C ^{0}F Kelvin mV ohm	0 Not Class'd
1, 247	SV	Temperature of external sensor (if sensor is RTD or TC)	32 33 35	^{0}C ^{0}F Kelvin	64 Temperature
2, 248	TV	Resistance of external sensor (if sensor is ohm or RTD)	37	ohm	85 Resistance
3, 249	QV	Voltage of external sensor (if sensor is mV or TC)	36	mV	0 Not Class'd
4	CJC Temp	Temperature of cold-junction sensor	32 33 35	^{0}C ^{0}F Kelvin	64 Temperature

Only Temperature (PV) allow changing of unit codes. Device Variable 4 automatically adapts the used Conductivity unit code.

6 DYNAMIC VARIABLES

Four Dynamic Variables are implemented.

Dyn var.	Meaning	Units
PV	Temperature, resistance or voltage of external sensor (depeding on selected sensor type).	degC, degF, Kelvin, ohm, mV
SV	Temperature of external sensor	degC, degF, Kelvin
TV	Resistance of external sensor (if sensor is ohm or RTD)	Ohm
QV	Voltage of external sensor (if sensor is mV or TC)	mV

For RTD sensors, the PV is derived from the sensor's resistance, using a polynomial equation. For thermocouples, the PV is derived from the millivolt input signal, using a combination of table look-up and linear interpolation, with compensation for the cold junction temperature.

7 STATUS INFORMATION

7.1 Device Status

The Field Device Status byte is contained in the second data byte in messages from the device. The following table defines the meaning of the different status bits.

Bit	Definition	Description
7	Device Malfunction	Is set if an electronic defect or memory defect is detected. This bit is set if a sensor break is detected.
6	Configuration Changed	Is set if a HART command results in writing new data to a configuration register.
5	Cold Start	Is set upon restart. It is reset for each master after responding to the first command from that specific master.
4	More Status Available	Is set if any of the Additional Device Status bits change status.
3	Loop Current Fixed	This bit is set if device is running in Fixed Current Mode (Command 40) or if Loop Current Signaling mode is turned off (e.g. in Multidrop Mode).
2	Loop Current Saturated	Is set if the output current associated to PV is capped by either the upper or lower current limits.
1	Non-Primary Variable Out of Limits	NOT USED
0	Primary Variable Out of Limits	Is set if PV is high or low limited.

Command #48 gives further detail. (See Section 7.3)

7.2 Extended Device Status

Extended Device Status is returned along with Additional Device Status by HART Command 48. Two bits are supported in this device.

Bit	Definition	Description
0	Maintenance Required	Is set if an electronic defect or memory defect is detected. This bit is set if a sensor break is detected.
1	Device Variable Alert	This bit is set if any Device Variable is simulated/fixed, or the environmental conditions are out of range. It will also be se if an electronic defect or memory defect is detected.
2	Critical Power Failure	Is set if the device detects that the power supply is not performing as expected.
3	Failure	Is set if an electronic defect, watchdog reset or if a memory defect is detected.
4	Out Of Specification	Is set if the environmental conditions are out of range

7.3 Additional Device Status (Command #48)

Command #48 returns 14 bytes of data, with the following status information:

Byte	Bit	Definition	Description
0	Device Specific Error Status Flags		
	7-1	NOT USED	
	0	Sensor Break	Is set if wire break is detected
1	Device Specific 0		
	7-0	NOT USED	
2	Device Specific 1		
	7-0	NOT USED	
3	Device Specific 2		
	7-0	NOT USED	
4	Device Specific 3		
	7-0	NOT USED	
5	Device Specific 4		
	7-0	NOT USED	
6	Extended Device Status		
	7-4	NOT USED	
	3	Failure	See Extended Device Status
	2	Critical Power Failure	See Extended Device Status
	1	Device Variable Alert	See Extended Device Status
	0	Maintenance Required	See Extended Device Status
7	NOT USED		
	7-0	NOT USED	
8	Standardized Status 0		
	7	Device Configuration Locked	NOT USED
	6	Electronic Defect	Is set in case of sensor break
	5	Environment Conditions out of	This bit is set if ambient temperatures is out of range.

		Range	
	4	Power Supply Conditions out of Range	NOT USED
	3	Watchdog Reset Executed	This bit is set in case of the watchdog resetting the device, in case of firmware running into a software dead-lock.
	2	Non-volatile Memory Defect	This bit is set if a problem with the system memory is detected.
	1	Device Variable Simulation Active	Is set if any device variable is being simulated, e.g. by in-factory system test.
	0	NOT USED	
9	Standardized Status 1		
	7-3	NOT USED	
	2	Event Notification Overflow	This bit is set if the internal processor becomes overworked, not able to execute all tasks given within the allowed time.
	1	Discrete Variable Simulation	NOT USED
	0	Simulation Active	NOT USED

NOT USED bits are always set to 0.

These status bits are updated several times each second. They are set by any failure detected by the periodic status update routine.

8 UNIVERSAL COMMANDS

8.1 Supported Universal Commands

All Universal Commands are mandatory and are supported by the device. Following Universal Commands are implemented:

0	Read Unique Identifier
1	Read Primary Variable
2	Read Loop Current And Percent Of Range
3	Read Dynamic Variables And Loop Current
6	Write Polling Address
7	Read Loop Configuration
8	Read Dynamic Variable Classifications
9	Read Device Variables with Status
11	Read Unique Identifier Associated With Tag
12	Read Message
13	Read Tag, Descriptor, Date
14	Read Primary Variable Transducer Information
15	Read Device Information
16	Read Final Assembly Number
17	Write Message
18	Write Tag, Descriptor, Date
19	Write Final Assembly Number
20	Read Long Tag
21	Read Unique Identifier Associated With Long Tag
22	Write Long Tag
38	Reset Configuration Changed Flag
48	Read Additional Device Status

8.2 Special notes on Universal Commands

Command #3: Returns PV, SV, TV and QV. This totals in 24 data bytes.
Command #9: This command supports up to 8 device variables. This totals in up to 69 data bytes, including the time stamp.
Command #14: Transducer and sensor serial numbers are not supported. The units code for limits and minimum span is equal to that of the Primary Variable.
Command #15: Write protect is not implemented, and Write Protect Code is therefore always returned as “251” (None). The unit code for Primary Variable range values is the same as is used for the Primary Variable.
Command #48: Returns 14 bytes of data.

9 COMMON-PRACTICE COMMANDS

9.1 Supported Common Practice Commands

The following common-practice commands are implemented:

34	Write Primary Variable Damping Value
35	Write Primary Variable Range Values
36	Set Primary Variable Upper Range Value
37	Set Primary Variable Lower Range Value
40	Enter/Exit Fixed Current Mode
42	Perform Device Reset
43	Set Primary Variable Zero
44	Write Primary Variable Units
47	Write PV Transfer Function
50	Read Dynamic Variable Assignment
60	Read Analog Channel And Percent Of Range
72	Squawk
520	Read Process Unit Tag
521	Write Process Unit Tag

9.2 Special notes on Universal Commands

Command #47: Supports only setting TFC Special Curve (4) or TFC Linear (0), enabling or disabling user specified lookup table (when running Ohm, Pot-Ohm or mV sensor type class, for RTD or TC always Linear).
Command #72: The device flashes with the red/green LED with a frequency of 5Hz.

10 TEMPERATURE DEVICE FAMILY COMMANDS

10.1 Supported Temperature Device Family Commands

The following temperature device family commands are implemented:

1024	Read Temperature Status
1025	Read Temperature Config
1026	Read Thermocouple Config
1152	Write Temp Probe Type
1155	Select Cold Junction Compensation Type
1156	Write Manual Cold Junction Temperature

11 DEVICE-SPECIFIC COMMANDS

The following device specific commands are implemented:

136	Read Min Max Measured
137	Reset Min Max Measured
138	Sensor Trim
139	Read Sensor Trim
142	Read Fixed Cjc Temp
143	Write Fixed Cjc Temp
144	Read Dummy Unit
145	Write 2w Cable Resistance
146	Read 2w Cable Resistance
147	Set Measurement Settings
148	Get Measurement Settings
149	Write Output Limits
150	Read Output Limits

151	Read Serial No String
152	Read Production Data
153	Write Sensor Config
154	Read Sensor Config
155	Write Offset Config
156	Read Offset Config
157	Start 2W Auto Compensation
158	Get 2W Auto Compensation Status
159	Factory Reset

11.1 Command 136 – Read Minimum And Maximum Measured Primary Values

This command returns the Minimum and Maximum measured value since startup/reset.
The values are of the currently selected unit.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..3	float	Minimum Measured Value
4..7	float	Maximum Measured Value

11.2 Command 137 – Reset Minimum And Maximum Measure Primary Values

This command resets the Minimum and Maximum measured value since startup/reset.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
-	-	No response bytes

11.3 Command 138 – Write Sensor Offset/Trim

This command is used to setup new values for the sensor offset and gain. The input values are stored must be of the currently selected temperature unit.

Request data frame

Byte	Format	Description
0..3	float	Sensor Offset Value
4	Bool	Reset

Response data frame

Byte	Format	Description
0..3	float	Sensor Offset Value

11.4 Command 139 – Read Sensor Offset/Trim

This command is used read the sensor offset and gain values. The returned value is represented in the currently selected temperature unit.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..3	float	Sensor Offset Value

11.5 Command 142 – Read Fixed CJC temperature

This command is used read the fixed CJC temperature. It is not curtain if the fixed CJC is selected. See Command 153.
The returned value represents the CJC temperature and is of the currently selected temperature unit.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..3	float	TC_fixedCJC_temp_conf

11.6 Command 143 – Write Fixed CJC temperature

This command is used to setup new values for the sensor offset and gain. It is not curtain if the fixed CJC is selected. See Command 153.
The input values are stored as a fixed CJC temperature and must be of the currently selected temperature unit.

Request data frame

Byte	Format	Description
0..3	float	TC_fixedCJC_temp_conf

Response data frame

Byte	Format	Description
0..3	float	TC_fixedCJC_temp_conf

11.7 Command 144 – Read Dummy

This command is used read a dummy byte. Can be used to check if device uses correct framing etc.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0	byte	Dummy

11.8 Command 145 – Write 2W cable resistance

This command is used to setup new values for 2W cable resistance.
The input values are stored as a 2W cable resistance and must be of unit ohm.

Request data frame

Byte	Format	Description
0..3	float	InputCableResistance

Response data frame

Byte	Format	Description
0..3	float	InputCableResistance

Response codes

Code	Description
0	No Command Errors
5	Too Few Data Bytes
13	Range Values Out Of Limits

11.9 Command 146 – Read 2W cable resistance

This command is used to read values for 2W cable resistance.
The returned value represents the 2W cable resistance and must be of unit ohm.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..3	float	InputCableResistance

11.10 Command 147 – Set Measurement settings

This command is used to setup new values for measurementSetting.
The input values are stored as measurementSetting and must be of type e_measurement_settings_t.

Request data frame

Byte	Format	Description
1	bitfield	measurementSetting

Response data frame

Byte	Format	Description
1	bitfield	measurementSetting

11.11 Command 148 – Get Measurement settings

This command is used to read values for measurementSetting.
The returned value represents the measurementSetting and must be of type e_measurement_settings_t (8-bit bitfield).

```
struct
{
    uint8_t WB_detect:1;
    uint8_t fast_response:1;
    uint8_t reserved2:1;
    uint8_t reserved3:1;
    uint8_t reserved4:1;
    uint8_t reserved5:1;
    uint8_t reserved6:1;
    uint8_t reserved7:1;
} e_measurement_settings_t;
```

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
1	bitfield	measurementSetting

11.12 Command 149 – Write Output limits

This command is used to setup new values for lower-, upper current limit and alarm value.
The input values are stored as the lower-, upper current limit and alarm value and is of unit mA.

Request data frame

Byte	Format	Description
0..1	uint16	OutputLowerCurrentLimit
2..3	uint16	OutputUpperCurrentLimit
4..5	uint16	OutMaAlarmSensorBreak

Response data frame

Byte	Format	Description
0..1	uint16	OutputLowerCurrentLimit
2..3	uint16	OutputUpperCurrentLimit
4..5	uint16	OutMaAlarmSensorBreak

Response codes

Code	Description
0	No Command Errors
5	Too Few Data Bytes
13	Values Out Of Limits

11.13 Command 150 – Read Output limits

This command is used read the output limit values for lower-, upper current limit and alarm value.
The returned value represents the lower-, upper current limit and alarm value and is of unit mA.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..1	uint16	OutputLowerCurrentLimit
2..3	uint16	OutputUpperCurrentLimit
4..5	uint16	OutMaAlarmSensorBreak

11.14 Command 151 – Read Serial No String

This command is used to read the serial number string.
The returned value represents the serial number string.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..23	uint8[24] ASCII	SerialNo string

11.15 Command 152 – Read Production Data

This command is used to read production data.
The returned value represents the HW version as a string, FW version and production date.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..23	uint8[24] ASCII	HWversion string,
24	uint8	u8_FW_VERSION_MAJOR
25	uint8	u8_FW_VERSION_MINOR
26	uint8	u8_FW_VERSION_BUILD
27	uint8	Prod_day
28	uint8	Prod_month
29..30	uint16	Prod_Year

11.16 Command 153 – Write Sensor Config

This command is used to write the sensor configuration.
The values are stored as the sensor type for RTD sensor (and CJC sensor), sensor type for TC sensor, sensor type class (RTD, TC, mV or Ohm), base value for RTD sensor and wire configuration for RTD and for TC.

Request and Response data frames

Byte	Format	Description
0	e_sensorTypeRTD_t	sensorTypeRTD_cfg
1	e_sensorTypeTC_t	sensorTypeTC_cfg
2	e_sensorTypeClass_t	sensorTypeClass
3..4	uint16	RTD_baseValue
5	e_WIRE_COMP_RTD_t	InputConnectionTypeRTD
6	e_WIRE_COMP_TC_t	InputConnectionTypeTC

```
typedef enum
{
   e_sensorTypeTC_0_B = 0,
   e_sensorTypeTC_1_E,
   e_sensorTypeTC_2_J,
   e_sensorTypeTC_3_K,
   e_sensorTypeTC_4_L,
   e_sensorTypeTC_5_N,
   e_sensorTypeTC_6_R,
   e_sensorTypeTC_7_S,
   e_sensorTypeTC_8_T,
   e_sensorTypeTC_9_U,
   e_sensorTypeTC_10_W3,
   e_sensorTypeTC_11_W5,
   e_sensorTypeTC_12_User_Spec,
} e_sensorTypeTC_t;
```

```
typedef enum
{
   e_sensorTypeRTD_0_Pt_A3850 = 0,
   e_sensorTypeRTD_1_Pt_A3902,
   e_sensorTypeRTD_2_Pt_A3916,
   e_sensorTypeRTD_3_NI_A1618,
   e_sensorTypeRTD_4_CU_A1428,
   e_sensorTypeRTD_5_Pt_A3920,
   e_sensorTypeRTD_6_User_Spec,
} e_sensorTypeRTD_t;


typedef enum
{
   e_SENSORTYPECLASS_RTD = 0,
   e_SENSORTYPECLASS_TC,
   e_SENSORTYPECLASS_MV,
   e_SENSORTYPECLASS_OHM,
   e_SENSORTYPECLASS_OHM_POT,
} e_sensorTypeClass_t;


typedef enum
{
   e_WIRE_COMP_RTD_WIRE2 = 0,
   e_WIRE_COMP_RTD_WIRE3,
   e_WIRE_COMP_RTD_WIRE4,
   e_WIRE_COMP_RTD_INTERNAL, // Used by TC CJC compensation
} e_WIRE_COMP_RTD_t;


typedef enum
{
   e_WIRE_COMP_TC_Wire2 = 0,
   e_WIRE_COMP_TC_Wire3,
   e_WIRE_COMP_TC_Fixed,
   e_WIRE_COMP_TC_Internal,
} e_WIRE_COMP_TC_t;
```

11.17 Command 154 – Read Sensor Config

This command is used to read the sensor configuration.
The returned value represents the sensor type for RTD sensor (and CJC sensor), sensor type for TC sensor, sensor type class (RTD, TC, mV or Ohm), base value for RTD sensor and wire configuration for RTD and for TC.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0	uint8	sensorTypeRTD_cfg
1	uint8	sensorTypeTC_cfg
2	uint8	sensorTypeClass
3..4	uint16	RTD_baseValue
5	uint8	InputConnectionTypeRTD
6	uint8	InputConnectionTypeTC

11.18 Command 155 – Write Offset Config

This command is used to write the sensor offset configuration.
The values are stored as the offset corrections for mV, ohm, Cjc and for primary temperature measurement.
e_active_correction_t could be used to clear Pt2 and/or Pt3 data if enum set to one_pt or two_pt.

Request data frame

Byte	Format	Description
0..3	float	mV_SensorOffsetCorrection
4..7	float	ohm_SensorOffsetCorrection
8..11	float	CjcTempOffsetConf
12..15	float	Pt1_f32_corrected
16..19	float	Pt1_f32_uncorrected
20..23	float	Pt2_f32_corrected
24..27	float	Pt2_f32_uncorrected
28..31	float	Pt3_f32_corrected
32..35	float	Pt3_f32_uncorrected
36	enum	e_active_temp_correction_t

Response data frame

Byte	Format	Description
0..3	float	mV_SensorOffsetCorrection
4..7	float	ohm_SensorOffsetCorrection
8..11	float	CjcTempOffsetConf
12..15	float	Pt1_f32_corrected
16..19	float	Pt1_f32_uncorrected
20..23	float	Pt2_f32_corrected
24..27	float	Pt2_f32_uncorrected
28..31	float	Pt3_f32_corrected
32..35	float	Pt3_f32_uncorrected
36	enum	e_active_temp_correction_t

```
typedef enum
{
   disabled = 0,
   one_pt,
   two_pt,
   three_pt,
} e_active_temp_correction_t;
```

11.19 Command 156 – Read Offset Config

This command is used to read the sensor offset configuration.
The returned value represents the offset corrections for mV, ohm, Cjc and for primary temperature measurement.
e_active_correction_t indicates which type of temperature correction is used.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0..3	float	mV_SensorOffsetCorrection
4..7	float	ohm_SensorOffsetCorrection
8..11	float	CjcTempOffsetConf
12..15	float	Pt1_f32_corrected
16..19	float	Pt1_f32_uncorrected
20..23	float	Pt2_f32_corrected
24..27	float	Pt2_f32_uncorrected
28..31	float	Pt3_f32_corrected
32..35	float	Pt3_f32_uncorrected
36	enum	e_active_temp_correction_t

11.20 Command 157 – Start 2W Auto Compensation

This command is used to initiate the 2W auto compensation. The RTD must be shorted before sending the command, so only wire resistance is measured by the transmitter.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
-	-	No request bytes

11.21 Command 158 – Get 2W Auto Compensation Status

This command is used to read the 2W auto compensation execution status.
The returned value represents the status. Returned value 1 means ongoing, returned value 0 means completed.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
0	bool	Compensate2WOnGoing

11.22 Command 159 – Factory Reset

This command sets user configuration back to settings when delivered from manufacturer, and restores all sensor corrections set by user.

Request data frame

Byte	Format	Description
-	-	No request bytes

Response data frame

Byte	Format	Description
-	-	No response bytes

12 Performance

12.1 Sampling Rates

Typical sampling rates are shown in the following table.

Sensor	Sampling rate
Temperature	Up to 20 per second
Ohm	Approx. 7 per second
mV	Approx. 7 per second
CJC	Approx. 7 per second
Internal temp	Approx. 1 per second

12.2 Power-Up

On power up, the transmitter runs through a startup initialization procedure, which takes approximately 3 seconds. During this period, the device will not be able to respond to HART commands, and the analog output is set at 3.5mA, and red LED is lightened up.
The first stable measurements are ready in less than 4 seconds, allowing valid Device Variable readouts. Fixed-current mode is cancelled upon startup / reset.

12.3 Reset

Command 42 - Perform Device Reset causes the device to reset its microprocessor. The resulting restart is identical to the normal power up sequence.

12.4 Self-Test

The FlexTop 2222 does not support command 41 – Self Test. Self-testing is performed periodically during normal operation.

12.5 Command Response Times

HART command response time depends on the command number issued and the internal state of the device. In order to ensure power consumption all time is below 3.5 mA, the response might be delayed up to 150 ms, due to other power consuming modules.

Generalization	Response times
Minimum	10 ms
Typical	50 ms
Maximum	150 ms

12.6 Busy and Delayed-Response

The transmitter may respond with "busy" status if a further command is received while self-test is underway or device is processing another command.
Delayed-response is responeded if a further command is received while device is processing another command.

12.7 Long Messages

The largest data field used is in the response to Command 9: 69 data bytes plus status bytes and header etc.

12.8 Non-Volatile Memory

FLASH is used to hold the device's configuration parameters. New data is written to this memory immediately on execution of a write command.

12.9 Modes

Fixed current mode is implemented, using Command 40. This mode is cleared by power loss or reset.

12.10 Burst Mode

This Field Device does not support Burst Mode.

12.11 Write Protection

Write-protection is not provided.

12.12 Catch Device Variable

This Field Device does not support Catch Device Variable.

12.13 Damping

Damping is standard, affecting only the loop current signal. Per firmware version 02.00.01 damping on the PV is not supported.

ANNEX. A CAPABILITY CHECKLIST

Manufacturer, model and revision	Baumer A/S, FlexTop2222, rev. 1
Device type	Transmitter
HART revision	7.0
Device Description available	Yes
Number and type of sensors	2 (one external, one internal)
Number and type of actuators	0
Number and type of host side signals	1: 4 - 20mA analog
Number of Device Variables	7
Number of Dynamic Variables	4
Mappable Dynamic Variables?	No
Number of common-practice commands	13
Number of device family commands	8
Number of device-specific commands	22
Bits of additional device status	8
Alternative operating modes?	No
Burst mode?	No
Write-protection?	No

ANNEX. B DEFAULT CONFIGURATION

Parameter	Default value
Lower Range Value	0
Upper Range Value	100
PV Units	degC
Sensor type	Pt100, α=0.385
Number of wires	2
Damping time constant	0 second
Fault-indication	Up-scale, 23mA
Number of response preambles	5

ANNEX. C REVISION HISTORY

A1. 2019-03-01 First revision 1.0

Document created.