User Manual

PTM/N SDI-12

Revisionshistory

Version	Datum	Autor	Description
1.0			Initialversion
1.3	06.06.2008	MM	Examples added
2.0	10.12.2008	RB	Instruction set expanded in according to FW1.2 and examples added.

Contents

1 Contact addresses

STS Sensor Technik Sirnach AG
Rütihofstrasse 8
8370 Sirnach
+41 71 969 49 29
+41 71 969 49 20
http://www.sts-ag.ch

STS GmbH
Poststrasse 7
71063 Sindelfingen
Germany
Tel: +49 (0)7031 204 9410
Fax: +49 (0)7031 204 9420

STS Italia
Via Gesu 5
20090 Opera (MI)
Italy
Tel: 0039 (0)2 5760 7073
Fax: 0039 (0)2 5760 7110
http://www.sts-italia.it

STS France
844, Route de la Caille
74350 Allonzier la Caille
France
Tel: 0033 450 08 48 15
Mobile: 0033 (0)6 0744 3806
Fax: 0033 450 67 02 43
http://www.stsfrance.fr

STS Great Britain Ltd.
Coppice House
Halesfield 7
Telford TF7 4NA
Great Britain
Tel: +44 (0) 1952 581093
Fax: +44 (0) 1952 581046
http://www.sts-gb.co.uk

2 Literature

[1] SDI-12 specifications http://www.sdi-12.org

3 Conventions/Abbreviations

P	Pressure
P_{ZP}	Zero-point pressure
P_N	Nominal pressure
ZP	Zero-point
T	Temperature
T_{ZP}	Start of temperature range
T_N	End of temperature range
DW	Data word
DB	Data byte
@	at

4 PTM/N/SDI-12

4.1 Functions

The PTM SDI-12 provides the following functions:

- Zero-point and nominal pressure recalibration
- SDI-12 interface
- Pressure units selectable
- Temperature units selectable
- Additional identification string

4.2 Technical Data

Please see in the datasheet

4.3 Safety instructions

Please see operating- and safety instructions

4.4 Installation

Please see operating- and safety instructions

4.5 Beginning of operation

4.6 Interface

An SDI-12 interface is used for communication with the PTM SDI-12. The following settings apply here:

Protocol:	SDI-12
Default address:	0 (factory settings)
Transfer rate:	1200 baud
Start bit:	1
Data bits:	7
Parity:	1 bit, even
Stop bit:	1

5 SDI-12

5.1 General information

SDI-12 is the acronym for "Serial Data Interface at 1200 Baud". SDI-12 is an asynchronous, ASCII, serial communications protocol that was developed for intelligent sensory instruments that typically monitor environmental data. These instruments are typically low-power (12 volt), are often used in remote locations, and usually communicate with a data logger or other data acquisition device. In this master-slave configuration, the data logger or data acquisition device typically acts as the master (SDI-12 Recorder and Interrogator) to the data monitoring instruments, which are the slaves (SDI-12 sensors). One master can communicate with multiple slaves, so the SDI-12 protocol requires that each device in the serial network be identified with an unique address, which is represented by a single ASCII character. Ensure that each sensor has an unique address before using in the network. Otherwise bus error will be occurring.

5.2 Transmit command

A transmit command is transmitted from a data logger or a controller to the sensor. It always consists of the address (0 … 9, A … Z, a … z) and the terminator ("!"). Most commands also have a command character (details see chapter 6).
The address character "?" can also be used. But in this case all sensors respond to the bus, which can lead to communication difficulties if more than one sensor is connected to the SDI-12 bus. For this reason, it must be ensured that only one sensor is connected to the SDI-12 bus.

5.3 Sensor response

The sensor response is carried out on a transmit command. It always consists of the address and the terminator (<CR><LF>). Most responses also contain data with or without a CRC-16.

6 SDI-12 Standard Commands

6.1 Standard commands according to specification SDI-12 V1.3

All sensors with an SDI-12 interface must support these commands. They are specified as mandatory in the SDI-12 specification V1.3 on web site http://www.sdi-12.org.

6.1.1 Break

A break is sent to wake up the sensors. In case of a break, all sensors must wake up so that they do not miss the commands that follow. A break is a constant bus level (spacing) for 12 ms.

6.1.2 Acknowledge Active

Forces the sensor to confirm its presence. This command has no command character.

Command:	a!	
Response:	a<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)

6.1.3 Send Identification Command

The sensor provides information on itself, enabling it to be identified.

Command:	aI!	
Response:	allccccccccmmmmmmvvv<ser><CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	ll	SDI-12 versions number ll = 13 (SDI-12 Version V1.3)
	cccccccc	Company name (8 characters) cccccccc = STS␣AG␣␣
	mmmm mm	Sensor model number (6 characters)
	vvv	Sensor version number vvv = 1.1 (Version 1.1 of sensor firmware)
	<ser>	Serial number (6 … 13 characters)

6.1.3.1 Example Send Identification Command

Command:	0I!
Response:	013STS AG 4900001.1654321<CR><LF>

6.1.4 Change Address Command

The sensor address is changed with this command. The address "0" is the default address, which must be set at the factory when shipping out the sensor. It is advisable to change the address over to another address during initial start-up of the sensor to avoid communication problems (address conflicts).

Command:	aAb!	
Response:	b<CR><LF>	
Explanation:	a	old sensor address (0, 1…9, A…Z, a…z)
	b	new sensor address (0, 1…9, A…Z, a…z)

6.1.4.1 Example Change Address Command

Changing the Address from "0" to "1"

Command: 0A1!
Response: 1<CR><LF>

Changing the Address from "1" to "0"

Command: 1A0!
Response: 0<CR><LF>

6.1.5 Address Query Command

This command queries the sensor address. It responds, regardless of which address it has. All sensors on the bus respond. This can result in communication problems if more than one sensor is connected to the SDI-12 bus.

Command:	?!	
Response:	a<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)

6.1.6 Start Measurement Command

This command informs the sensor that it is to start its measurements. With this command all measuring channels (pressure and temperature) are measured. It responds with the number of measurements and the time until these are ready to be picked up. The sensor itself does not send any measuring results. Data must then be picked up with the "D" command. If the sensor has the measuring results available before the specified time expires, then it sends a Service Request (see chapter 6.1.9) to the data logger.

Command:	aM!	Requesting of measured values without CRC-16 checksum
	aMC!	Requesting of measured values with CRC-16 checksum
Response:	atttn<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	ttt	Time in seconds until measuring results are available
	n	No. of measuring results (0…9)
		n = 2 (One pressure and one temperature measured value)

6.1.6.1 Example Start Measurement Command

Please see example chapter 6.1.7.

6.1.7 Send Data Command

This command requests the measured data from the sensor. It is required to pick up several measured data from the sensor. The command is carried out on an "M", "C" or "V" command of the data logger.

Command:	aD0!	
Response:	a<val><CR><LF>	Response when a measurement without a CRC-16 checksum is required
	a<val>ccc<CR><LF>	Response of the sensor when a measurement with a CRC-16 checksum is required

Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	<val>	Measured values (1 to 7 numbers, max. 9 characters including sign (+,-) and decimal point per measured value).
	ccc	CRC-16 checksum (3 characters)

6.1.7.1 Examples Get Measurement

Command:	0M!	(start measurement)
Response:	00012<CR><LF>	(2 measurements ready in 1 second)
Response:	0<CR><LF>	(Service Request)
Command:	0D0!	(send data)
Response:	0+0.012-1.3<CR><LF>	(e.g. +0.012bar, -1.3°C)

Command:	0M1!	(start measurement)
Response:	00011<CR><LF>	(2 measurements ready in 1 second)
Response:	0<CR><LF>	(Service Request)
Command:	0D0!	(send data)
Response:	0+0.012<CR><LF>	(e.g. +0.012bar)

Command:	0M2!	(start measurement)
Response:	00011<CR><LF>	(2 measurements ready in 1 second)
Response:	0<CR><LF>	(Service Request)
Command:	0D0!	(send data)
Response:	0-1.3<CR><LF>	(e.g. -1.3°C)

Command:	0C!	(start measurement)
Response:	000102<CR><LF>	(2 measurements ready in 1 second)
Command:	0D0!	(send data)
Response:	0+0.012-1.3<CR><LF>	(e.g. +0.012bar, -1.3°C)

6.1.8 Additional Measurements Command

This command is for sensors which support several measuring variables (e.g. pressure, temperature). The "M1" command starts the pressure measurement and the "M2" command the temperature measurement. It responds with the number of measurements and the time until these are ready to be picked up. The sensor itself does not send any measuring results. Data must then be picked up with the "D" command. If the sensor has the measuring results available before the specified time expires, then it sends a Service Request (see chapter 6.1.9) to the data logger.

Command:	aM1!	Start pressure measurement, requesting of measured values without CRC-16 checksum
	aM2!	Start temperature measurement, requesting of measured values without CRC-16 checksum
Response:	atttn<CR><LF>	

Command:	aMC1!	Start pressure measurement, requesting of measured values with CRC-16 checksum
	aMC2!	Start temperature measurement, requesting of measured values with CRC-16 checksum
Response:	atttnccc<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	ttt	Time in seconds until measuring results are available
	n	No. of measuring results n = 1 (One pressure or temperature measured value)
	ccc	CRC-16 checksum (3 characters)

The commands aM3! … aM9! and aMC3! … aMC9! are implemented but have no function behind. In this case the sensor responds always "a0000<CR><LF>" for a "M" command and "a0000ccc<CR><LF> for a "MC" command, which is interpreted as a fault message by the data logger.

6.1.8.1 Example Additional Measurement Command

Please see example chapter 6.1.7.

6.1.9 Service Request (from sensor to data logger)

This command is initiated by the sensor if it has measuring results available earlier than assumed. An "M" or "V" command has been sent by the data logger depending on the process. As a result, it informs the data logger, that it now has measuring results available.

Command: "M" or "V" command, depending on process
Response: a<CR><LF>

Explanation: a Sensor address (0, 1…9, A…Z, a…z)

6.1.10 Start Concurrent Measurement Command

Starts a measurement in the sensor, while another sensor is still busy with its measurements (also "C" command). With this command all measuring channels (pressure and temperature) are measured. The sensor does not send a Service Request when it is finished with the measurements. The measuring results must then be picked up with the "D" command. Measurements can be triggered in several sensors with this command.

Command:	aC!	Requesting of measured values without CRC-16 checksum
Response:	atttnn<CR><LF>	
Command:	aCC!	Requesting of measured values with CRC-16 checksum
Response:	atttnnccc<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	ttt	Time in seconds until measuring results are available
	nn	No. of measuring results
		nn = 02 (One pressure and one temperature measured value)
	ccc	CRC-16 checksum (3 characters)

6.1.10.1 Example Concurrent Measurement Command

Please see example chapter 6.1.7.

6.1.11 Start Verification Command

This command is implemented in according to the SDI-12 standard but has no function behind.

Command:	aV!	
Response:	a0000<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)

Sending the "D" command to pick up the data is not necessary but allowed. The response after a "D" command is always a<CR><LF>

6.1.12 Additional Concurrent Measurements Command

Measurements can be triggered in several sensors with this command. It starts a measurement in the sensor, while another sensor is still busy with its measurements (also "C" command). This command is intended for sensors which support several measuring variables (e.g. pressure, temperature). The "C1" command starts the pressure measurement and the "C2" command the temperature measurement. The sensor responds with the number of measurements and the time until these are ready. The sensor itself does not send any measuring results. The sensor does not send a Service Request when it is finished with the measurements. The measuring results must then be picked up with the "D" command.

Command:	aC1!	Start pressure measurement, requesting of measured values without CRC-16 checksum

	aC2!	Start temperature measurement, requesting of measured values without CRC-16 checksum
Response:	atttnn<CR><LF>	
Command:	aCC1!	Start pressure measurement, requesting of measured values with CRC-16 checksum
	aCC2!	Start temperature measurement, requesting of measured values with CRC-16 checksum
Response:	atttnnccc<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	ttt	Time in seconds until measuring results are available
	nn	No. of measuring results n = 01 (One pressure or temperature measured value)
	ccc	CRC-16 checksum (3 characters)

The commands aC3! … aC9! and aCC3! … aCC9! are implemented but have no function behind. In this case the sensor responds always "a00000<CR><LF>" for a "C" command and "a00000ccc<CR><LF> for a "CC" command, which is interpreted as a fault message by the data logger.

6.1.13 Continuous Measurement Command

This command is implemented in according to the SDI-12 standard but has no function behind.

Command:	aR0...aR9!	Requesting of measured values without CRC-16 checksum
Response:	a<CR><LF>	
Command:	aRC0...aRC9!	Requesting of measured values with CRC-16 checksum
Response:	accc<CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	ccc	CRC-16 checksum (3 characters)

6.2 CRC-16 Checksum

Details see SDI-12 standard V1.3 chapter 4.4.12

6.3 SDI-12 extended commands

6.3.1 General information

This chapter describes extended commands which extend the standard command set. All extended commands are marked with an X as the first command character.
Settings made by extended commands are volatile and not automatically saved. For permanent storage use the save command (chapter 6.3.12). Otherwise after a power up, the settings will be lost.

6.3.2 Setting pressure unit

Causes the sensor to calculate measurement results in another unit. The measured value is output in the selected pressure unit in future.

Command:	aXPnn!	
Response:	ann<CR><LF>	Sensor has switched to new unit
	a0000<CR><LF>	Has not recognized the new unit (fault message)
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	nn	Index of unit (see table below)

Table of units:	Index	Unit
	00	factory settings
	01	bar
	02	mbar
	03	mWC / mH2O
	04	psi
	05	ftWC
	06	inH2O

Settings made by extended commands are volatile and not automatically saved. For permanent storage use the save command (chapter 6.3.12). Otherwise after a power up, the settings will be lost.

6.3.2.1 Example setting pressure unit

Initial condition:	address:	0
	unit:	mWC

Command:	0XP01!	(change unit to bar)
Response:	001<CR><LF>	(index of unit 01 = bar)

6.3.3 Reading pressure unit

Causes the sensor to output the selected unit.

Command: aXP!

Response: ann<CR><LF>

Explanation: a Sensor address (0, 1…9, A…Z, a…z)

nn Index of unit (see table below)

Table of units:

Index	Unit
01	bar
02	mbar
03	mWC / mH2O
04	psi
05	ftWC
06	inH2O

6.3.4 Setting recalibration value of zero-point pressure

Use this command for recalibration of zero-point pressure or reset the zero-point pressure recalibration to default. The recalibration value <value> has to be in the adjusted unit. For reset to default set the recalibration value to default zero-point pressure in the actually adjusted unit.

Further information for calculation recalibration value, recalibration procedure and recalibration examples please see chapter 7.

Command: aXZZ<value>!

Response: a<value><CR><LF>

Explanation: a Sensor address (0, 1…9, A…Z, a…z)

<value> recalibration value in the adjusted unit (max. 8 characters including sign and dezimal point)

Settings made by extended commands are volatile and not automatically saved. For permanent storage use the save command (chapter 6.3.12). Otherwise after a power up, the settings will be lost.

6.3.5 Reading recalibration value of zero-point pressure

Initiate the sensor to output the recalibration value of the zero-point pressure.

Command: aXZZ!

Response: a<value><CR><LF>

Explanation: a Sensor address (0, 1…9, A…Z, a…z)

<value> recalibration value in the adjusted unit (max. 8 characters including sign and dezimal point)

6.3.6 Setting recalibration value of nominal pressure

Use this command for recalibration of nominal pressure or reset the nominal pressure recalibration to default. The recalibration value <value> has to be in the adjusted unit. For reset to default set the recalibration value to default nominal pressure in the actually adjusted unit.
Further information for calculation recalibration value, recalibration procedure and recalibration examples please see chapter 7.

Command:	aXZF<value>!	
Response:	a<value><CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	<value>	recalibration value in the adjusted unit (max. 8 characters including sign and dezimal point)

Settings made by extended commands are volatile and not automatically saved. For permanent storage use the save command (chapter 6.3.12). Otherwise after a power up, the settings will be lost.

6.3.7 Reading recalibration value of nominal pressure

Initiate the sensor to output the recalibration value of the nominal pressure.

Command:	aXZF!	
Response:	a<value><CR><LF>	
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	<value>	recalibration value in the adjusted unit (max. 8 characters including sign and dezimal point)

6.3.8 Setting temperature unit

Causes the sensor to calculate measurement results in another unit. The measured value is output in the selected temperature unit in future.

Command:	aXTn!	
Response:	an<CR><LF>	Sensor has switched to new unit
	a0000<CR><LF>	Has not recognized the new unit (fault message)
Explanation:	a	Sensor address (0, 1…9, A…Z, a…z)
	n	Index of unit (see table below)

Table of units:	Index	Unit
	0	factory settings
	1	°C
	2	°F
	3	K

Settings made by extended commands are volatile and not automatically saved. For permanent storage use the save command (chapter 6.3.12). Otherwise after a power up, the settings will be lost.

6.3.8.1 Example setting temperature unit

Initial condition:	address:	0
	unit:	°C

Command:	0XT2!	(change unit to °F)
Response:	02<CR><LF>	(index of unit 2 = °F)

6.3.9 Reading temperature unit

Causes the sensor to output the selected unit.

Command:	aXT!	
Response:	an<CR><LF>	
Explanation:	a	Sensor (0, 1…9, A…Z, a…z)
	n	Index of unit (see table below)

Table of units:	Index	Unit
	1	°C
	2	°F
	3	K

6.3.10 Setting user identification string

Writes an identification string in the sensor. The identification string has a maximum length of 16 characters. Using this identification string, the customer can name the sensor as desired. It can use this to store the measurement location of the sensor in the sensor.

Command:	aXI<string>!	
Response:	a<string><CR><LF>	
Explanation:	a	Sensor address (0, 1 - 9, A - Z, a - z)
	<string>	Identification string (1 … 16 printable characters)

Settings made by extended commands are volatile and not automatically saved. For permanent storage use the save command (chapter 6.3.12). Otherwise after a power up, the settings will be lost.

6.3.11 Reading user identification string

Reads the identification string of the sensor. The identification string has a maximum length of 16 characters.

Command:	aXI!	
Response:	a<string><CR><LF>	
Explanation:	a	Sensor address (0, 1 - 9, A - Z, a - z)
	<string>	Identification string (max. 16 characters)

6.3.12 Saving settings in flash memory

Causes the sensor to save all settings in the internal flash memory. This is necessary to save the changed settings for the long term. This also retains the settings in case the power supply is interrupted. They can be changed again at any time. If the settings are not saved, they will be lost when the sensor is restarted (power supply is interrupted).

Command:	aXF!	
Response:	a<CR><LF>	
Explanation:	a	Sensor address (0, 1 - 9, A - Z, a - z)

7 Recalibration

Normally STS sensors have an excellent long-term stability and are calibrated by STS. Occasionally a recalibration could be necessary. For this situation it is possible to recalibrate the zero-point and nominal pressure by use of a linear recalibration function. It is not possible to recalibrate errors for more than 5%FS. In this case, please send the sensor back to STS for recalibration. The recalibration function is not intended for taring.

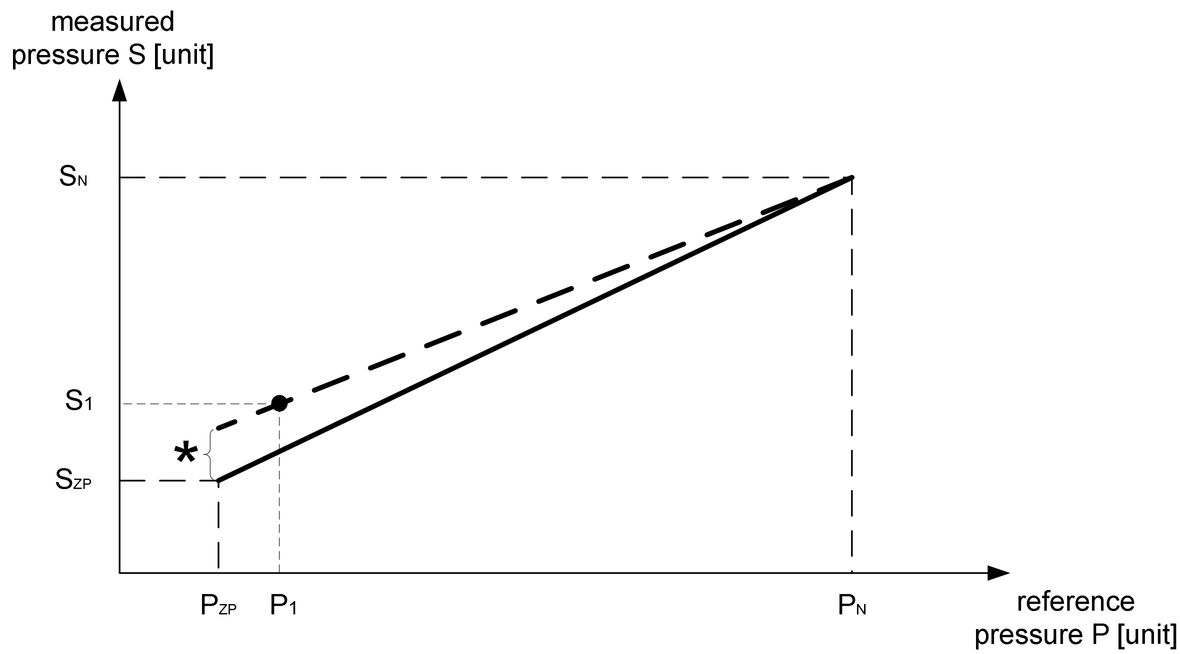
For recalibration we recommend using a pressure reference which is 5 times more precisely than the sensor. Ensure that reference pressure is stable.

7.1 Recalibration procedure

There are three different ways for recalibration: zero-point recalibration, nominal pressure recalibration. In the following each procedure is explained step by step.

7.1.1 Zero-point pressure recalibration

1. Read out the actual recalibration value of nominal pressure -> $PUserCalFullscale_{old}$ (chapter 6.3.7)
2. Read out the actual recalibration value of zero-point pressure -> $PUserCalZero_{old}$ (chapter 6.3.4)
3. Set reference pressure (near by zero-point pressure) -> P_1
4. Read out the pressure from the sensor -> S_1 (with "M" and "D" Command)
5. Calculate the recalibration value if necessary (see below)
6. Write the recalibration value to the sensor (chapter 6.3.4)
7. Set zero-point reference pressure
8. Read out the pressure from the sensor (with "M" and "D" Command)
9. Calculate the error (repeat procedure if necessary)
10. Save the recalibration value (chapter 6.3.12)



*	recalibration value	P_N	nominal pressure ref.	S_N	nominal pressure sensor
(solid line)	ideal characteristic	P_{ZP}	zero-point pressure ref.	S_{ZP}	zero-point pressure sensor
(dashed line)	actual characteristic	P_1	reference pressure	S_1	measured pressure @ P_1

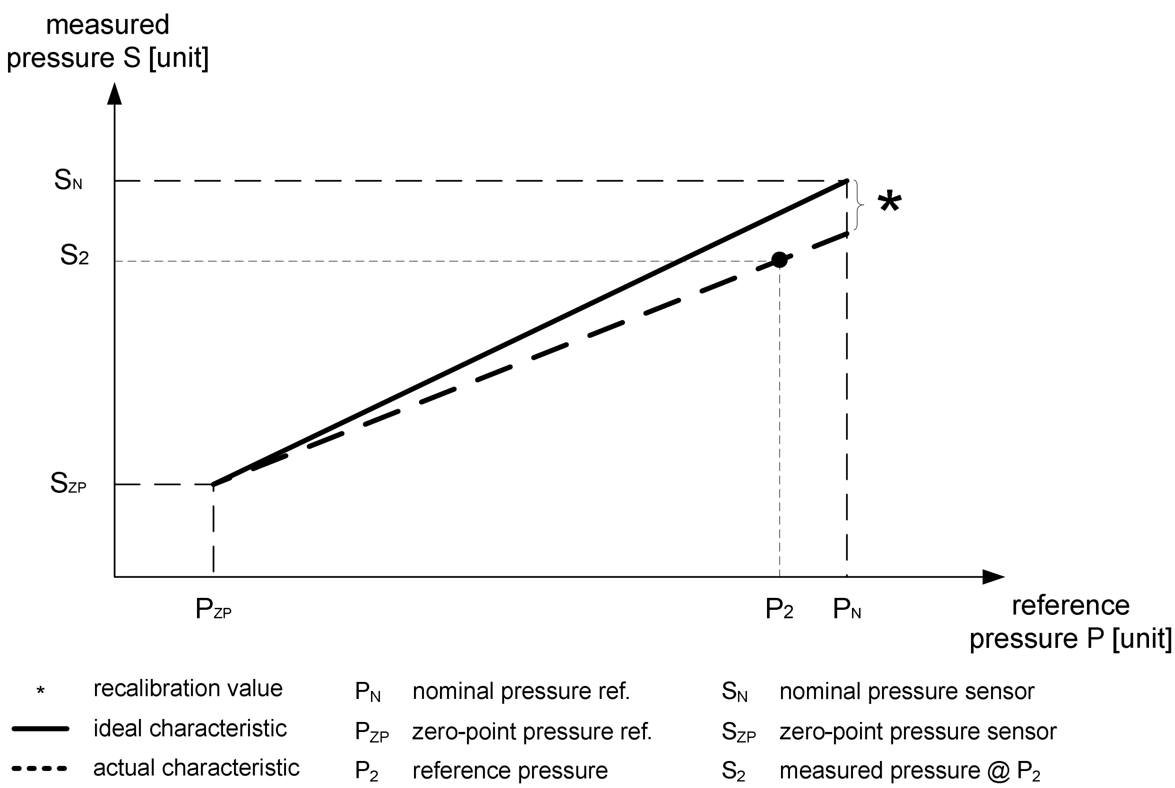
Calculation of zero-point pressure recalibration value

$$G_{PTM} = \frac{P_N - P_{ZP}}{(PUserCalFullscale_{old} - PUserCalZero_{old})} \qquad G = \frac{(S_N - S_1)}{(P_N - P_1)}$$

$$PUserCalZero_{new} = PUserCalZero_{old} + \frac{[S_1 - (P_1 - P_{ZP}) * G]}{G_{PTM}}$$

7.1.2 Nominal pressure recalibration

1. Read out the actual recalibration value of nominal pressure -> $PUserCalFullscale_{old}$ (chapter 6.3.7)
2. Read out the actual recalibration value of zero-point pressure -> $PUserCalZero_{old}$ (chapter 6.3.4)
3. Set reference pressure (near by nominal pressure) -> P_2
4. Read out the pressure from the sensor -> S_2 (with "M" and "D" Command)
5. Calculate the recalibration value if necessary (see below)
6. Write the recalibration value to the sensor (chapter 6.3.4)
7. Set nominal reference pressure
8. Read out the pressure from the sensor (with "M" and "D" Command)
9. Calculate the error (repeat procedure if necessary)
10. Save the recalibration value (chapter 6.3.12)



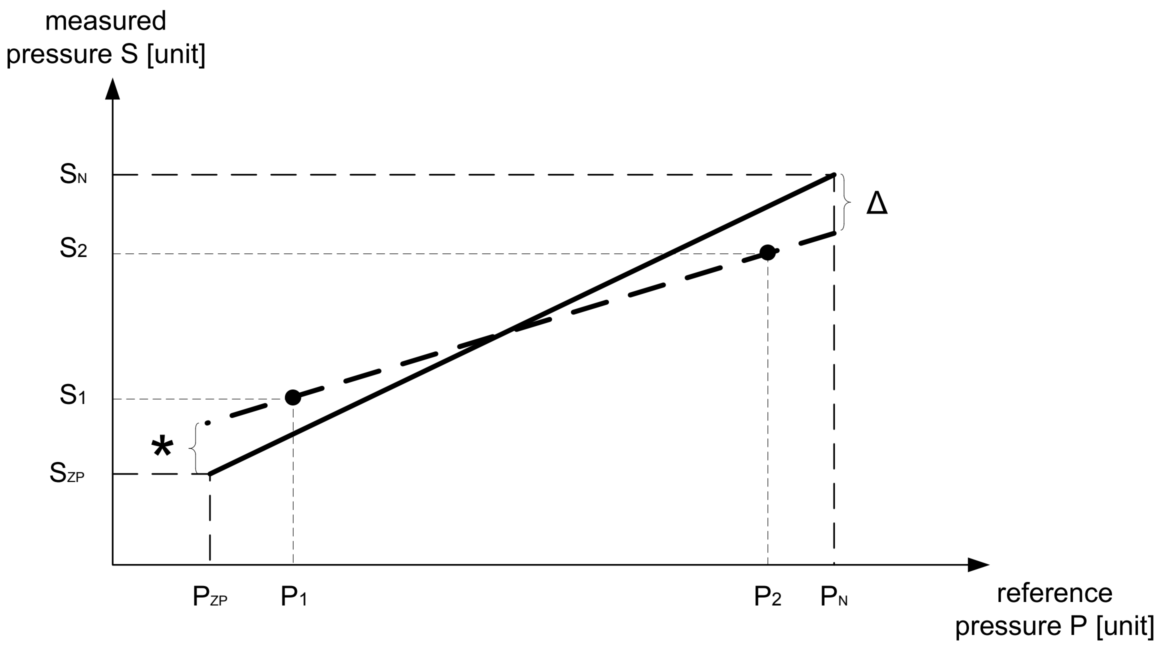
Calculation of nominal pressure recalibration value

$$G_{PTM} = \frac{P_N - P_{ZP}}{(PUserCalFullscale_{old} - PUserCalZero_{old})} \qquad G = \frac{(S_2 - P_{ZP})}{(P_2 - P_{ZP})}$$

$$PUserCalFullscale_{new} = PUserCalFullscale_{old} - \frac{[P_N - S_2 - (P_N - P_2) * G]}{G_{PTM}}$$

7.1.3 Zero-point and nominal pressure recalibration

1. Read out the actual recalibration value of nominal pressure -> $PUserCalFullscale_{old}$ (chapter 6.3.7)
2. Read out the actual recalibration value of zero-point pressure -> $PUserCalZero_{old}$ (chapter 6.3.4)
3. Set reference pressure (near by zero-point pressure) -> P_1
4. Read out the pressure from the sensor -> S_1 (with “M” and “D” Command)
5. Set reference pressure (near by nominal pressure) -> P_2
6. Read out the pressure from the sensor -> S_2 (with “M” and “D” Command)
7. Calculate the recalibration value if necessary (see below)
8. Write the recalibration value to the sensor (chapter 6.3.4)
9. Set zero-point reference pressure
10. Read out the pressure from the sensor (with “M” and “D” Command)
11. Set nominal reference pressure
12. Read out the pressure from the sensor (with “M” and “D” Command)
13. Calculate the error (repeat procedure if necessary)
14. Save the recalibration value (chapter 6.3.12)



*	recalibration value zero-point	P_N	nominal pressure ref.	S_N	nominal pressure sensor
Δ	recalibration value nominal	P_{ZP}	zero-point pressure ref.	S_{ZP}	zero-point pressure sensor
———	ideal characteristic	P_1	reference pressure	S_1	measured pressure @ P_1
- - - -	actual characteristic	P_2	reference pressure	S_2	measured pressure @ P_2

Calculation of zero-point and nominal recalibration value

$$G_{PTM} = \frac{P_N - P_{ZP}}{(PUserCalFullscale_{old} - PUserCalZero_{old})} \qquad G = \frac{(S_2 - S_1)}{(P_2 - P_1)}$$

$$PUserCalZero_{new} = PUserCalZero_{old} + \frac{[S_1 - (P_1 - P_{ZP}) * G]}{G_{PTM}}$$

$$PUserCalFullscale_{new} = PUserCalFullscale_{old} - \frac{[P_N - S_2 - (P_N - P_2) * G]}{G_{PTM}}$$

8 Unit conversion

8.1 Pressure unit conversion table

Unit	bar	mbar	mWC / mH2O	psi	ftWC	inH2O
bar	1	1000	10.1968	14.5033	33.456	401.606
mbar	0.001	1	0.0101968	0.014503	0.033456	0.401606
mWC / mH2O	0.09807	98.07	1	1.422338	3.281299	39.38550
psi	0.06895	68.95	0.7030693	1	2.306785	27.69069
ftWC	0.02989	29.89	0.3047823	0.433503	1	12.00401
inH2O	0.00249	2.49	0.0253900	0.036113	0.083305	1

8.2 Temperature unit conversion

Kelvin -> Grad Celsius:

T[°C] = T[K] – 273.15

Grad Celsius -> Kelvin:

T[K] = T[°C] + 273.15

Grad Fahrenheit -> Grad Celsius:

T[°C] = 5/9 * T[F] – 160/9

Grad Celsius -> Grad Fahrenheit:

T[°F] = 9/5 * T[°C] + 32

Kelvin -> Grad Fahrenheit:

T[°F] = 9/5 * (T[K] - 273.15) + 32

Grad Fahrenheit -> Kelvin:

T[K] = (5/9 * T[°F] – 160/9) + 273.15