



# Inspired by **temperature**

## **Data Communication**

**Manual**

**huber**



# Data Communication



# Table of Contents

V2.1.0en/10.10.19

<b>1</b>	<b>Introduction .....</b>	<b>7</b>
<b>1.1</b>	<b>Safety .....</b>	<b>7</b>
1.1.1	Symbols used for Safety Instructions.....	7
<b>2</b>	<b>Application .....</b>	<b>8</b>
<b>3</b>	<b>General Rules of PB commands .....</b>	<b>8</b>
<b>4</b>	<b>Properties of PB Commands .....</b>	<b>9</b>
<b>5</b>	<b>Timing of PB commands .....</b>	<b>10</b>
<b>5.1</b>	<b>Timing rules .....</b>	<b>10</b>
<b>5.2</b>	<b>Timing recommendations.....</b>	<b>10</b>
<b>6</b>	<b>General Description of a PB command.....</b>	<b>11</b>
<b>6.1</b>	<b>Structure.....</b>	<b>11</b>
<b>6.2</b>	<b>Rule violations .....</b>	<b>11</b>
<b>7</b>	<b>Special Description of PB commands .....</b>	<b>12</b>
<b>8</b>	<b>PB command examples.....</b>	<b>28</b>
<b>9</b>	<b>PB commands with greater value range and higher accuracy ....</b>	<b>30</b>
<b>9.1</b>	<b>Examples .....</b>	<b>31</b>
<b>10</b>	<b>PB package command.....</b>	<b>32</b>
<b>10.1</b>	<b>Structure.....</b>	<b>32</b>
<b>10.2</b>	<b>Structure with greater value range and higher accuracy.....</b>	<b>33</b>
<b>10.3</b>	<b>Configuration .....</b>	<b>33</b>
<b>10.4</b>	<b>Examples .....</b>	<b>34</b>
<b>11</b>	<b>Modbus .....</b>	<b>37</b>
<b>11.1</b>	<b>General description.....</b>	<b>37</b>
<b>11.2</b>	<b>Data types .....</b>	<b>38</b>
<b>11.3</b>	<b>Error handling .....</b>	<b>38</b>
<b>11.4</b>	<b>Specific description of function codes.....</b>	<b>38</b>
11.4.1	Public Funktionscodes.....	39
11.4.1.1	Read Holding Registers (FC 0x03) .....	39
11.4.1.2	Write Single Holding Register (FC 0x06) .....	40
11.4.2	User-Defined Functioncodes.....	42
11.4.2.1	Communication test (FC 0x41).....	42
11.4.2.2	Query PB variable (FC 0x42) .....	42
11.4.2.3	Changing and querying a PB variable (FC 0x43).....	43
11.4.2.4	Querying a PB package command (FC 0x44).....	44
11.4.2.5	Changing and querying a PB package command (FC 0x45) .....	46
<b>12</b>	<b>OPC-UA.....</b>	<b>47</b>
<b>12.1</b>	<b>What is OPC UA? .....</b>	<b>47</b>
<b>12.2</b>	<b>Description of function.....</b>	<b>47</b>
12.2.1	What can the Huber OPC server do? .....	47
12.2.2	Number of clients.....	47
12.2.3	Speed / Polling interval .....	47
12.2.4	Subscriptions.....	47
<b>12.3</b>	<b>Prerequisites.....</b>	<b>48</b>




12.3.1	Network .....	48
12.3.2	Certificates .....	48
<b>12.4</b>	<b>Setup mode .....</b>	<b>48</b>
12.4.1	Pilot ONE® Configuration .....	48
12.4.1.1	<i>Setting the time</i> .....	49
12.4.1.2	<i>Ethernet set-up</i> .....	49
<b>12.5</b>	<b>Setting a Huber OPC Server .....</b>	<b>49</b>
12.5.1	Start/stop .....	49
12.5.2	Encryption .....	50
12.5.3	Configuration .....	50
12.5.4	Exchanging certificates.....	50

# 1 Introduction

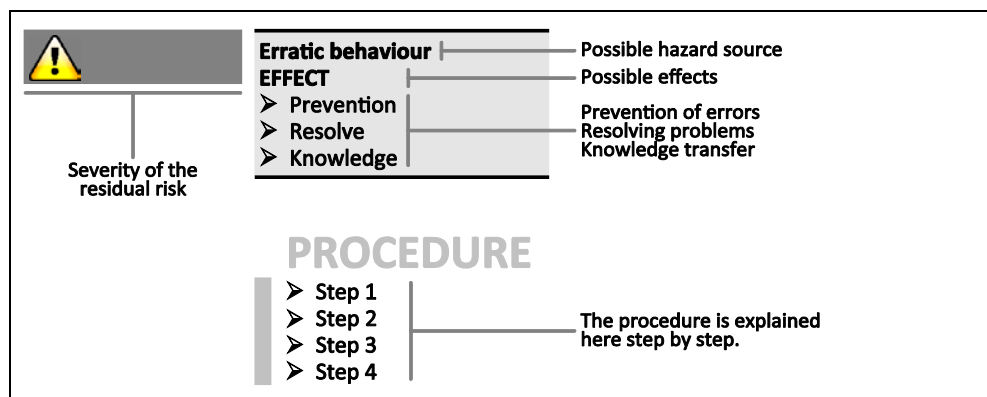
## 1.1 Safety

### 1.1.1 Symbols used for Safety Instructions

Safety instructions are indicated by the below combinations of pictograms and signal words. The signal word describes the classification of the residual risk when disregarding the operation manual.

 <b>DANGER</b>	Denotes an immediate hazardous situation that will result in death or serious injuries.
 <b>WARNING</b>	Denotes a general hazardous situation that may result in death or serious injuries.
 <b>CAUTION</b>	Denotes a hazardous situation that can result in injury.
<b>NOTE</b>	Denotes a situation that can result in property material damage.
<b>INFORMATION</b>	Denotes important notes and usable hints.

Explanation  
Safety information and  
procedure



The safety information in this manual is designed to protect the responsible body, operator and the equipment from damage. Safety instructions appear at the beginning of each chapter and before instructions. First inform yourself about any residual risks due to misuse before you start an operation.

## 2 Application

Huber offers various interfaces for its thermostats, depending on the operating unit. Using these, it is possible to communicate with the thermostat. They include RS232, RS485, Ethernet or USB. It is possible to use the commands described here via all these interfaces.

This manual describes the use of the PB commands. PB commands are implemented in all thermostats with graphic display (Unistat Control/Pilot, CC-Pilot and Pilot ONE®). PB commands can be combined into a single PB package command if several PB commands are addressed simultaneously in a single data transfer.

This manual also describes how to use the Modbus protocol for thermostats with Pilot ONE®. However, only the Modbus TCP mode via the Ethernet interface is supported.

The E-grade "OPC-UA" also supports the OPC-UA protocol. Several OPC clients can connect to the OPC server on the pilot ONE and read and write the PB commands.

### INFORMATION

Apart from the PB commands described in this manual, there are also the PP and LAI commands. These will also continue to be supported. However, for new developments, it is advisable to use the PB commands because of their simple, standardized programming.

### INFORMATION

If several thermostats are to be controlled at one RS485 bus, it is not possible to use the PB commands. For this, you should use the LAI commands, which are not described further here.

### INFORMATION

When communicating via the RS232 interface, the following parameters must be set:  
Baud rate: 9600 (standard, with Com.G@te also other baud rates possible); Start bits: 1; Data bits: 8; Parity: none; Stop bits: 1; Handshake: None

### INFORMATION

The Ethernet communication with Pilot ONE uses TCP (Transmission Control Protocol), Port 8101. The interfaces used must meet the specifications of the generally accepted standards.

### INFORMATION

Information on communication with Profibus Gateway SE (Second Edition) or 3E (Third Edition) can be found in the corresponding manual.

### INFORMATION

The older controller generations of Unistat Control / Pilot and CC-Pilot do not support all commands described (commands to addresses 0-0x47 are supported).  
It may be necessary to update the thermostat software to be able to use all the described functions.

## 3 General Rules of PB commands

The PB commands are intended for data exchange with precisely two participants. These commands have a simple structure and transmission is not secured via checksums, which is why the communication software is also very simple.

The motivation for the development was to keep the character strings to be transmitted as simple as possible. In this way, the work required for further data processing can be kept to a minimum. To ensure secure operation, the thermostat response must be received and evaluated. With Ethernet, data security is also provided by the TCP/IP protocol.

### INFORMATION

Some of the commands described in the following may be blocked. These can be released by installing the suitable E-grade. You will find further information on our website at [www.huber-online.com](http://www.huber-online.com). A 30-day evaluation version is available free of charge.

In general, it can be said that all settings that are changed by a data command will not be permanently saved and will be reset to the last manually entered value after any interruption in the power supply.



## 4 Properties of PB Commands

- The PB commands are available for devices with Unistat Pilot, CC-Pilot or Pilot ONE® (devices from 2005 onward).
- PB commands can be used in point-to-point transmissions.
- PB commands consist of character strings.
- The master / slave principle applies. The thermostat is always the slave. This means that the thermostat only ever transmits a command if a request has been received from the master. The slave always sends a response if the received character string is recognized as a valid command. No slave is allowed to transmit without being requested.
- Figures are shown as hexadecimal figures in two's complement. The ASCII characters '0' ... '9' and 'A' ... 'F' are used for number transmission. Consequently, 2 numbers per byte must be transmitted.
- The structure of the PB commands is very simple. A command is always 10 characters long.
- The timing of the data transmission is determined by the response principle (see chapter on Timing).
- The PB commands are organized in the form of a table. An address is assigned to every variable. A variable can be changed or requested by contacting the relevant address.
- If an address is contacted which is not defined or not released (cf. E-grade), the command is answered with the same address and the value "7FFF".
- A response is only sent to syntactically correctly received PB commands.
- A command sets the current value and returns the current value of the requested variable. If the time progression of a variable is to be observed, the command must be sent repeatedly.
- Only one single variable value can be queried per command. It is not possible to string several values together.
- The table can be expanded if necessary; old contents are not changed.

## 5 Timing of PB commands

### 5.1 Timing rules

The software protocols are structured in such a way that very simple timing rules can be used.

The data flow within a command should not be interrupted. Breaks of more than 100 ms between the characters of a command will lead in the thermostat to the abortion of the incoming command. Sometimes the master's operating system can be responsible for such problems. An oscilloscope or perhaps a data logger installed on a fast monitoring PC can help with checking this.

The protocols are structured in such a way that a correct inquiry must always be answered by a response from the thermostat. Before a new command can be sent, the system must wait for the complete response from the thermostat. If a new command is transmitted beforehand, this is rejected. Once the master has received the response from the thermostat, the next command can be transmitted immediately. Normally, only a few tenths of a second pass before the thermostat's response. The typical response time is below 300 ms.

If no response has been received after this time, the command should be repeated again. For security, the time until the repeat can also be made slightly longer. In general, we recommend waiting at least a second for the response in order not to risk unnecessary timeouts even in less favorable situations.

#### INFORMATION

If Com.G@te® or Web.G@te® is used, the response time may be slightly longer, and may exceed 300 ms. In this case, please adjust your timeout condition accordingly.

### 5.2 Timing recommendations

If the speed is pushed to the maximum, a data throughput is obtained which is very high for a temperature control unit. Even if the master and slave can cope with this, you should think about which renewal rate is really required. You will find that it is almost always sufficient to stay in the range of a few seconds. It does not make sense technically to stuff the hard disk with redundant data or to take computing time away unnecessarily from the thermostat.

If very large data volumes have to be transmitted, as is occasionally necessary for process documentation, the repeat rate can if necessary be adjusted to the relevant signal dynamic. This means that some signals are requested more frequently than others.

When we talk about "large data volumes", this ultimately means, in this context, levels of 10 or 20 bytes/s. But together with the typically long running times for temperature control processes, considerable volumes can then occur. At 20 bytes/s, a megabyte is reached in just 14 hours or so.

## 6 General Description of a PB command

### 6.1 Structure

A PB command consists of 10 characters. A PB command is structured as follows:

{mttvvv<CR><LF>

{	Start character (ASCII code 0x7B): identifies the PB command.
m	Sender ID: Master request 'M' (ASCII code 0x4D) or slave response 'S' (ASCII Code 0x53) to designate the direction. The automation system (PC, SPC, PCS, ...) is always the master. Its request is designated by an 'M'. The thermostat is always the slave. The thermostat's response is designated by an 'S'.
tt	Address (corresponds to table place): 2 characters (1 byte in hexadecimal representation).
vvv	Variable value: 4 characters (2 bytes as hexadecimal string) If a variable is only to be requested and not changed, the variable value "****" should be sent. The response for temperatures that are not available is -151.00 °C. This corresponds to the character series "C504".
<CR>	The Carriage Return symbol (ASCII code 0x0D)
<LF>	The Line Feed symbol (ASCII code 0x0A)

### 6.2 Rule violations

What do we mean by this?

Imprecise programming of a command string or timing that is too tight. Random and occasional deviations caused by transmission errors generally also lead to rule violations, but are not caused by the software.

How are rule violations handled?

In principle, it can be said: If the commands are not recognized, characters are missing or are added, the processing is terminated, there is no response. Not even a general command, such as that meaning "not understood" is sent, because the slave cannot be aware whether it is a bus communication or a point to point connection. The bus sovereignty could therefore just have been requested by another device.

## 7 Special Description of PB commands

The command table is shown in the following. Special notes on the variable type, the resolutions and the values are provided.

Notes on the table:

Address	The address is given in each case in decimal and in hexadecimal form (preceded by 0x) and corresponds to the place in the table.
Variable	Brief designation for documentation purposes
Description	Brief description of the variable
Type	R: Only read access to the variable is granted R / W: Read and write access is granted to the variable
LSB	The smallest unit that can be represented "0.01 °C" means, for example, that a variable value of 1 corresponds to a temperature of 0.01 °C or the value 100 corresponds to a temperature of 1.00 °C.
Data type	Integer: All values in the specified value range can occur. The value range indicates whether the conversion of the hexadecimal number to a decimal number has to be unsigned (value range between 0 and 65535) or signed (value range between -32768 and 32767). Peculiarity of temperatures: If a value range is specified between -15111 and 50000, the number is to be interpreted as a signed number. However, if the decimal number is smaller than -15111, it must be interpreted as an unsigned number (a hexadecimal value between 0x8000 and 0xC4F8 corresponds to a decimal value between 32768 and 50424). To make things simpler for thermostats with a temperature range above 300 °C, values can always be interpreted as unsigned values (as no temperature below 0 °C will occur) and the values of all other thermostats are signed values. Bit field: The value must be considered bit by bit. Every bit has a different significance.
Value range	The range that the variable value can assume (decimal representation). With a value range of -15111 ... 50000 and an LSB of 0.01 °C temperatures of -151.11 °C to 500.00 °C can therefore be processed. For bit fields, the following applies: With write procedures, only the bits agreed in each case can be filled with 1. With read procedures, only the bits agreed in each case can be assigned a value, all other bits are undefined.
E-grade	Various commands are available depending on which E-grade was activated for a device. Commands associated in the table with an E-grade are also available for a superior E-grade, as the various E-grades are based on each other (Basic, Exclusive, Professional, Explore). For example, the E-grade Professional includes all features found in E-grade Exclusive. E-grade Explore is offered only for Unistats. You can purchase DV-E-grade if you need some PB command variables. This activates all PB command variables. <b>Note:</b> The assignment of various PB command variables to E-grades has changed over time. E-grade specifications apply to devices with Pilot ONE® controller and the latest software version. Older-generation controller devices with Unistat Control / Pilot and CC-Pilot can only access a small part of the PB command variables. All other commands can be activated using DV-E-grade. Please contact the technical service for more information.

Some variables can be useless depending on the thermostat if the necessary sensors or actors are missing.

Address (hex)		Variable	Description	
00		vSP	Setpoint, temperature controller	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
<p>The setpoint is used by the temperature controller. With internal regulation, the setpoint applies for the internal temperature, with process regulation it applies for the process temperature. Please note: The setpoint specification can be overwritten by other setpoint indicators (e.g. temperature control program, analog 4...20 mA interface, ...).</p>				

Address (hex)		Variable	Description	
<b>01</b>		<b>vTI</b>	<b>Internal temperature</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Basic

From the point of view of the temperature control device, the current temperature of the thermal fluid which is flowing to the application. The internal temperature is sometimes also called the flow, jacket or bath temperature.

Address (hex)		Variable	Description	
<b>02</b>		<b>vTR</b>	<b>Return temperature</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Explore

From the point of view of the temperature control device the current temperature of the thermal fluid which is flowing back to the temperature control device. Not all temperature control devices measure this temperature. This sensor is typically installed on Unistats.  
If there is no sensor, the value -15100 is returned.

Address (hex)		Variable	Description	
<b>03</b>		<b>vpP</b>	<b>Pump pressure (absolute)</b>	
Type	LSB	Data type	Value range	E-grade
R	1 mbar	Integer	0 ... 32000	Basic

Absolute pump pressure at the pressure sensor in the thermal fluid circuit of the thermostat (only if there is a pressure sensor).  
The return value must be reduced by 1000 mbar if a relative pressure (the difference to the atmospheric pressure) is to be processed.

Address (hex)		Variable	Description	
<b>04</b>		<b>vPow</b>	<b>Current power</b>	
Type	LSB	Data type	Value range	E-grade
R	1 W	Integer	-32767 ... 32767	Explore

The currently resulting power of the thermostat in watts.  
The value is negative when the thermostat is cooling and positive when the thermostat is heating. It corresponds to the power that is made available to the application. The shaft power of the internal standard circulating pump is not taken into account. The cooling power is calculated indirectly via the current working point of the compressor. The heating power is calculated via the nominal voltage on the rating plate.  
For precision: With requirements for precision, the power values should be used more for a relative determination, if necessary after evaluation on a "calibration heating". The value provided must be evaluated with caution – it is an estimate. In comparable working points, the achieved reproducible precision is better than the absolute precision. To obtain reliable evaluations, regular comparative measurements are recommended.  
See also command 0x6E (vPowHi).

Address (hex)		Variable	Description	
<b>05</b>		<b>vError</b>	<b>Error report</b>	
Type	LSB	Data type	Value range	E-grade
R/W	1	Integer	-32768 ... 1	Basic

The number of the error report that appeared first.  
If 0 is returned, no error has occurred and the device is ready for use; otherwise, the nature of the error can be determined by the number.  
Send the value 1 to the thermostat to delete existing messages.

Address (hex)		Variable	Description	
<b>06</b>		<b>vWarn</b>	<b>Warning message</b>	
Type	LSB	Data type	Value range	E-grade
R/W	1	Integer	-32768 ... 1	Basic

The number of the most recent warning message to appear.  
A thermostat might need your attention if a number not equal 0 is returned. No current warnings exist, if 0 is returned; otherwise the number indicates the type of warning. Send the value 1 to the thermostat to delete existing messages.

Address (hex)		Variable	Description	
<b>07</b>		<b>vTE</b>	<b>Process temperature (Lemosa)</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Basic

The current process temperature is returned.  
Strictly speaking, the measured value of the Pt100 sensor which is connected to the LEMOSA connector is returned. Typically the processor sensor is connected here.  
If no sensor is connected, the value -151 °C is returned.

Address (hex)		Variable	Description	
<b>08</b>		<b>vIntMove</b>	<b>Actual value setting, Internal temperature</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Explore

With this, an alternative measured value outside the thermostat can be specified for the internal controller.  
The protective devices nonetheless use the internal Pt100 sensor. The command must be used in connection with command 0x18 (vCITM), otherwise a preset value is not taken into consideration. Monitoring is possible using command 0x0A (vStatus1, Bit 10).

Address (hex)		Variable	Description	
<b>09</b>		<b>vExtMove</b>	<b>Setting, Process temperature</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Explore

With process control, the process measured value can be specified for temperature control with this.  
The command must be used in connection with command 0x19 (vCETM) otherwise a preset value is not taken into consideration. Monitoring is possible using command 0x0A (vStatus1, Bit 11).

Address (hex)		Variable	Description	
<b>0A</b>		<b>vStatus1</b>	<b>Status of the thermostat</b>	
Type	LSB	Data type	Value range	E-grade
R	---	Bit field	---	Basic

Query current thermostat status.  
The returned value is a bit field. The individual bits must be considered independently of each other.

- Bit 0 Temperature control operating mode: 1: active / 0: inactive
- Bit 1 Circulation operating mode: 1: active / 0: inactive
- Bit 2 Refrigerator compressor: 1: switched on / 0: switched off
- Bit 3 Temperature control mode "Process control": 1: active / 0: inactive
- Bit 4 Circulating pump: 1: switched on / 0: switched off
- Bit 5 Cooling power available: 1: available / 0: not available
- Bit 6 Tkeylock: 1: active / 0: inactive
- Bit 7 PID parameter set, temperature controller : 1: Automatic mode / 0: Expert mode
- Bit 8 Error: 1: Error occurred / 0: no error
- Bit 9 Warning: 1: New warning occurred / 0: No new warning
- Bit 10 Mode for setting the internal temperature (see Address 8): 1: active / 0: inactive
- Bit 11 Mode for setting the external temperature (see Address 9): 1: active / 0: inactive
- Bit 12 DV E-grade: 1: activated / 0: not activated
- Bit 14 Restart electronics / Power failure (\*): 1: No new start / 0: New start
- Bit 15 Freeze protection (not available on all devices): 1: active / 0: inactive

(\*) This bit is reset to 0 when the control is restarted. If the variable vStatus1 is read for the first time after a restart, this bit has not been set. In all further reading procedures, this bit will send back a 1. In this way, an unexpected restart can be recognized.

Address (hex)		Variable	Description	
<b>0B</b>		<b>vBDPos</b>	<b>Control blow-down valve</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	-32700 ... 32700	Basic
<p>Specify blow-down valve position and query current position.            To be able to use this function, a special blow-down hardware must be provided.            Typical positions blow-down slide valve:            Position 0: Thermoregulation (normal state)            Position 4500: Blow-down            Position 8266: Keep            Position 2666: Empty            Homing is started from the current position if a value smaller than 0 is passed. The stepping motor moves the specified number of steps in the direction of the mechanical stop and then sets its actual position to the value 0. The transferred value must be great enough for the mechanical stop to be reached. A value of -10000 must be passed to ensure complete closure.            The response of the thermostat shows the current position of the blow-down valve.</p>				

Address (hex)		Variable	Description	
<b>0C</b>		<b>vBDHeat</b>	<b>Release blow-down valve heating</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
<p>Release heating for blow-down valve.            The release is monitored via Watchdog. If the release is not sent again to the thermostat within 10 s, the heating is switched off.            To be able to use this function, a special blow-down hardware with integrated heating must be provided.</p>				

Address (hex)		Variable	Description	
<b>0F</b>		<b>vNiv</b>	<b>Fill level</b>	
Type	LSB	Data type	Value range	E-grade
R	0.1%	Integer	-1 ... 1000	Basic
<p>Fill level of the thermostat.            This function is only available if an analog fill level display is integrated into your thermostat. If the value -1 is returned, then a problem has occurred with the fill level measurement.</p>				

Address (hex)		Variable	Description	
<b>12</b>		<b>vAutoPID</b>	<b>PID Parameter, automatic temperature controller</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
<p>Set and query currently used temperature controller parameter set.            The temperature controller works with the PID parameter set in automatic or in expert mode, as selected. The value 1 means that the controller is working in automatic mode; 0 means that the controller is working in expert mode. If you would like to work in expert mode, the individual control parameters must be entered first (see Address 0x1D – 0x25).</p>				

Address (hex)		Variable	Description	
<b>13</b>		<b>vTmpMode</b>	<b>Temperature control mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Exclusive
<p>Set and query the temperature control mode of the thermostat.            0: Temperature control mode internal            1: Temperature control mode process (cascade control)</p>				

Address (hex)		Variable	Description	
<b>14</b>		<b>vTmpActive</b>	<b>Temperature control</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
<p>Start or stop temperature control of the thermostat or query current status.            0: Temperature control not active            1: Temperature control active</p>				

Address (hex)		Variable	Description	
<b>15</b>		<b>vCompAuto</b>	<b>Compressor operating mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 2	Basic
<p>The compressor operating mode can be set and queried.</p> <p>0: The automatic compressor system is active. The thermostat decides independently whether the compressor is currently needed for temperature control and switches it on and off as required.</p> <p>1: The compressor is always switched on as long as the temperature control is active.</p> <p>2: The compressor is always switched off, irrespective of the thermostat operating mode.</p>				

Address (hex)		Variable	Description	
<b>16</b>		<b>vCircActive</b>	<b>Circulation</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
<p>Start or stop thermostat circulation or query current status. This operating mode is typically needed to set up the system.</p> <p>0: Circulation operating mode not active</p> <p>1: Circulation operating mode active</p> <p>Note: If the temperature control is active, circulation is also carried out, but circulation operating mode is not active.</p>				

Address (hex)		Variable	Description	
<b>17</b>		<b>vKeyLock</b>	<b>Operating lock</b>	
Type	LSB	Data type	Value range	E-grade
R / W	---	Bit field	---	Basic
<p>Activate, deactivate or query operating lock at Pilot.</p> <p>Bit 0: Operating lock active or inactive</p> <p>0: Operating lock inactive.</p> <p>1: Operating lock active. Manual operation of the thermostat via Pilot is not possible.</p> <p>Bit 1: Watchdog behavior</p> <p>0: Watchdog inactive.</p> <p>1: Activate Watchdog for 30 s. If Bit 1 is not reset within 30 s, the operating lock is automatically cancelled. This can be used to permit manual operation again if communication with the thermostat is interrupted for any reason.</p>				

Address (hex)		Variable	Description	
<b>18</b>		<b>vCITM</b>	<b>Internal temperature actual value setting mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	--	Bit field	---	Explore
<p>Activate, deactivate or query operating mode for setting the internal temperature.</p> <p>Bit 0: Operating mode active or inactive</p> <p>0: Mode not active.</p> <p>1: Mode active. Instead of the internal temperature, the temperature controller uses for control the current content of Variable 0x08 (vIntMove) to calculate the control variable. At the same time, when this bit is set for the first time, a Watchdog is activated which reacts after 30 s. By regularly setting the Variable 0x08 (vIntMove), it is possible to prevent the Watchdog from being triggered.</p> <p>Bit 1: Watchdog behavior</p> <p>Defines the behavior of the thermostat if a data communication problem is determined.</p> <p>0: The thermostat issues the warning -2129, deactivates the actual value setting and continues to operate temperature control by using the thermostat's internal sensor.</p> <p>1: The thermostat issues the error message -328 and changes to standby mode (temperature control is stopped).</p>				



Address (hex)		Variable	Description	
<b>19</b>		<b>vCETM</b>	<b>Process temperature actual value setting mode</b>	
Type	LSB	Data type	Value range	E-grade
R /W	--	Bit field	---	Explore
<p>Activate, deactivate or query operating mode for setting the process temperature.</p> <p>Bit 0: Operating mode active or inactive  0: Mode not active.  1: Mode active. Instead of the process temperature, the temperature controller uses for control the current content of Variable 0x09 (vExtMove) to calculate the control variable. At the same time, when this bit is set for the first time, a Watchdog is activated which reacts after 30 s. By regularly setting the Variable 0x09 (vExtMove) it is possible to prevent the Watchdog from being triggered.</p> <p>Bit 1: Watchdog behavior  Defines the behavior of the thermostat if a data communication problem is determined.  0: The thermostat issues the warning -2130, deactivates the actual value setting, changes from process to internal control (see vTmpMode, Variable 19) and continues to operate temperature control.  1: The thermostat issues the error message -329 and changes to standby mode (temperature control is stopped).</p>				

Address (hex)		Variable	Description	
<b>1A</b>		<b>vICE</b>	<b>Freeze protection</b>	
Type	LSB	Data type	Value range	E-grade
R /W	1	Integer	0 ... 1	Basic
<p>Activate, deactivate or query freeze protection of the thermostat.</p> <p>0: Freeze protection not active  1: Freeze protection active  Please note that freeze protection is not available for all thermostats.</p>				

Address (hex)		Variable	Description	
<b>1B + 1C</b>		<b>vSNRL + vSNRH</b>	<b>Serial number</b>	
Type	LSB	Data type	Value range	E-grade
R	1	Integer	0 ... 65535	Basic
<p>Query the serial number of the thermostat.</p> <p>Because the serial number can have larger values than 65535 but only 2 bytes per variable are available, the serial number is made up of 2 data words (Low + High) each with 2 bytes into a 4 byte value.</p> <p>Variable 0x1B (vSNRL) contains the low value bytes, Variable 0x1C (vSNRH) contains the high value bytes.</p> <p>See also chapter 9 »PB commands with greater value range and higher accuracy«.</p>				

Address (hex)		Variable	Description	
<b>1D</b>		<b>vKpInt</b>	<b>Kp of the internal controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	1	Integer	0 ... 32000	Basic
<p>Change and query amplification of the PID controller with internal regulation.</p> <p>A change affects the complete controller (P, I and D element).</p> <p>This parameter is only used if expert mode has been activated (see vAutoPID, Variable 0x12).</p>				

Address (hex)		Variable	Description	
<b>1E</b>		<b>vTnInt</b>	<b>Tn of the internal controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	0.1 s	Integer	0 ... 32000	Basic
<p>Change and query reset time of the PID controller with internal regulation.</p> <p>A change only affects the I element of the controller. If this Variable is set at 0, the I element of the controller is deactivated.</p> <p>This parameter is only used if expert mode has been activated (see vAutoPID, Address 0x12).</p>				

Address (hex)		Variable	Description	
<b>1F</b>		<b>vTvInt</b>	<b>Tv of the internal controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	0.1 s	Integer	0 ... 32000	Basic

Change and query hold-back time of the PID controller with internal regulation.  
A change only affects the D element of the controller.  
This parameter is only used if expert mode has been activated (see vAutoPID, Address 0x12).

Address (hex)		Variable	Description	
<b>20</b>		<b>vKpJack</b>	<b>Kp of the jacket controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	1	Integer	0 ... 32000	Exclusive

Change and query amplification of the PID controller with jacket control.  
A change affects the complete controller (P, I and D element).  
The parameter is only used with process control if expert mode has been activated (see vAutoPID, Address 0x12).  
The jacket controller is the subordinate controller if process control is activated.

Address (hex)		Variable	Description	
<b>21</b>		<b>vTnJack</b>	<b>Tn of the jacket controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	0.1 s	Integer	0 ... 32000	Exclusive

Change and query reset time of the PID controller with jacket control.  
A change only affects the I element of the controller.  
The jacket controller is essentially the same controller as the internal controller, but as a subordinate controller in the process cascade control. The reason for the distinction is that, with process control it is necessary to assign different parameters to the controller of the internal temperature than with internal regulation. This also makes sense, because no I element is necessary for the subordinate controller with process control.  
If this variable is set at 0, then the I element of the controller is deactivated. This is also the normal situation.  
The parameter is only used with process control (see vTmpMode, Variable 0x13) if expert mode has been activated (see vAutoPID, Address 0x12).

Address (hex)		Variable	Description	
<b>22</b>		<b>vTvJack</b>	<b>Tv of the jacket controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	0.1 s	Integer	0 ... 32000	Exclusive

Change and query hold-back time of the PID controller with jacket control.  
A change only affects the D element of the controller.  
Normally, this value is 0.  
The parameter is only used with process control if expert mode has been activated (see vAutoPID, Address 0x12).

Address (hex)		Variable	Description	
<b>23</b>		<b>vKpProc</b>	<b>Kp of the process controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	0.01	Integer	0 ... 32000	Exclusive

Change and query amplification of the PID controller with process control.  
A change affects the complete controller (P, I and D element).  
The parameter is only used with process control if expert mode has been activated (see vAutoPID, Address 0x12).

Address (hex)		Variable	Description	
<b>24</b>		<b>vTnProc</b>	<b>Tn of the process controller</b>	
Type	LSB	Data type	Value range	E-grade
R /W	0.1 s	Integer	0 ... 32000	Exclusive

Change and query reset time of the PID controller with process control.  
A change only affects the I element of the controller.  
If this Variable is set at 0, then the I element of the controller is deactivated.  
The parameter is only used with process control (see vTmpMode, Address 0x13) if expert mode has been activated (see vAutoPID, Address 0x12).

Address (hex)		Variable	Description	
<b>25</b>		<b>vTvProc</b>	<b>Tv of the process controller</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1 s	Integer	0 ... 32000	Exclusive
Change and query hold-back time of the PID controller with process control. A change only affects the D element of the controller. The parameter is only used with process control (see vTmpMode, Address 0x13) if expert mode has been activated (see vAutoPID, Address 0x12).				

Address (hex)		Variable	Description	
<b>26</b>		<b>vnP</b>	<b>Pump speed</b>	
Type	LSB	Data type	Value range	E-grade
R	1 / min	Integer	0 ... 32000	Basic
Query current pump speed. Please note that the current pump speed is not available with all thermostats.				

Address (hex)		Variable	Description	
<b>2C</b>		<b>vTKwIn</b>	<b>Cooling water entry temperature</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Explore
Current measured value of cooling water entry temperature.				

Address (hex)		Variable	Description	
<b>2D</b>		<b>vpKw</b>	<b>Cooling water pressure</b>	
Type	LSB	Data type	Value range	E-grade
R	1 mbar	Integer	0 ... 32000	Explore
Current measured value of cooling water pressure.				

Address (hex)		Variable	Description	
<b>2E</b>		<b>vPowCon</b>	<b>Power supply conditions</b>	
Type	LSB	Data type	Value range	E-grade
R / W	--	Bit field	---	Explore
With some temperature control devices, the user must inform the thermostat of the power supply values. This is normally done using a menu entry. This manual setting is recreated here. These variables must be used with the devices in question. Entering this information in devices that do not require it is ignored. Bit 0 - 2: Setting of supply voltage and frequency with switchable power supply: 000 No entry made / undefined 001 400 V / 50 Hz 010 400 V / 60 Hz 011 460 V / 60 Hz 100 480 V / 60 Hz Bits 8 - 9: Consideration of maximum power load by preliminary fuse. The device reduces the maximum electrical power consumption in such a way that the indicated maximum currents are not overrun. This function is only available in some devices with 230 V power supply. 00 No entry made / undefined 01 16A 10 13A 11 10A				

Address (hex)		Variable	Description	
<b>30</b>		<b>vMinSP</b>	<b>Minimum setpoint</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
This minimum setpoint is used by the temperature controller. It limits the setpoint range of the internal and process controller downward.				

Address (hex)		Variable	Description	
<b>31</b>		<b>vMaxSP</b>	<b>Maximum setpoint</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
This maximum setpoint is used by the temperature controller. It limits the setpoint range of the internal and process controller upward.				

Address (hex)		Variable	Description	
<b>33</b>		<b>vNivHi</b>	<b>Upper level limit</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1%	Integer	0 ... 1000	Basic
Setting the fill level upper limit. A higher fill level will generate a message (optional facility)				

Address (hex)		Variable	Description	
<b>34</b>		<b>vNivLo</b>	<b>Lower level limit</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1%	Integer	0 ... 1000	Basic
Setting the fill level lower limit. A lower fill level will generate a message (optional facility)				

Address (hex)		Variable	Description	
<b>35</b>		<b>vNivCont</b>	<b>Setting level output</b>	
Type	LSB	Data type	Value range	E-grade
R / W	--	Bit field	---	Basic
<p>Setting the switching direction for level contacts (optional facility).</p> <p>These are two digital outputs which can be set optionally as opener or closer. One output reports an over-level, the other an under-level.</p> <p>Bit 0: 0: Under-level output as opener 1: Under-level output as closer</p> <p>Bit 1: 0: Over-level output as opener 1: Over-level output as closer</p>				

Address (hex)		Variable	Description	
<b>3A</b>		<b>vTProc</b>	<b>Process temperature</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Exclusive
<p>The actual value of the temperature used by the process controller is returned.</p> <p>If no sensor is attached and the value is also not defined by a command or the analog input of the Com.G@te, the value -151 °C is returned.</p>				

Address (hex)		Variable	Description	
<b>3C</b>		<b>vStatus2</b>	<b>Status of the thermostat</b>	
Type	LSB	Data type	Value range	E-grade
R	---	Bit field	---	Basic
<p>Query current thermostat status.</p> <p>The returned value is a bit field. The individual bits must be considered independently of each other.</p> <p>Bit 0 Control: 1: The process controller currently specifies the control parameter / 0: The internal controller currently specifies the control parameter</p> <p>Bit 1-4 Active cooling controller (for internal use)</p> <p>Bit 5 The drip tray is full and needs to be emptied (pump seal leak) (only with special equipment)</p> <p>Bit 6 The VPC has completed its reference point approach and you can start the pump. Message -4137 is displayed on the Pilot ONE® if an attempt is made to start the pump while bit 6 returns the value 0.</p> <p>Bit 7 Venting mode: 1: active / 0: inactive</p> <p>Bit 8 In some devices (see bit 9), Pilot ONE automatically detects whether there is still air in the system in venting mode. 1: Venting successful, this operating mode can be terminated. 0: Ventilation not yet complete.</p> <p>Bit 9 In some devices, Pilot ONE automatically recognizes when the venting process can be terminated. 1: Venting is monitored, the state of the venting process can be checked with bit 8. 0: Venting is not monitored, the user must decide for himself when the venting can be completed. Bit 8 has no meaning in this case.</p>				

Address (hex)		Variable	Description	
<b>3D</b>		<b>vDistFeed</b>	<b>Disturbance feedforward</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1 W	Integer	-32767 ... 32767	Explore
<p>With foreseeable load changes, the thermostat can be informed, before the temperature actually changes, that it should adjust its power by the current value. In this way, it is possible to minimize the effects of disturbance variables which the controller can only detect after a certain downtime at the control probe.</p> <p>A positive value of the feedforward value increases the heating power or reduces the cooling capacity, a negative value behaves conversely.</p>				

Address (hex)		Variable	Description	
<b>3E</b>		<b>vpPIIn</b>	<b>Pressure in return (absolute)</b>	
Type	LSB	Data type	Value range	E-grade
R	1 mbar	Integer	0 ... 32000	Basic
<p>Absolute pressure in the return flow of the thermal fluid circuit of the thermostat (only with optional measurement device).</p>				

Address (hex)		Variable	Description	
<b>3F</b>		<b>vBIDown</b>	<b>Status Blow-Down</b>	
Type	LSB	Data type	Value range	E-grade
R / W	--	Bit field	---	Basic
<p>Status of the optional blow-down device.</p> <p>Bit 0 Compressed air OK</p> <p>Bit 1 Overtemperature status OK</p> <p>Bit 2 Operating mode OK (Standby)</p> <p>Bit 3 Internal temperature is OK (&lt; 70°C)</p> <p>Bit 4 Pressure in hydraulic circuit (pump pressure) is OK (&lt; 300mbar)</p> <p>Bit 5 Valve position OK</p> <p>Bit 8 Switch solenoid valve or blow-down process on / off</p> <p>With write access, only Bit 8 is evaluated.</p>				

Address (hex)		Variable	Description	
<b>40</b>		<b>vWD1</b>	<b>Watchdog (fault)</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1 s	Integer	0 ... 150	Basic
<p>Watchdog timer for monitoring data communication.</p> <p>If the timer is set to a value not equal to zero, the Watchdog is active. The variable must be regularly assigned the required time limit, otherwise the thermostat will generate a fault after expiry of the time.</p> <p>If the variable is assigned the value zero, the Watchdog is deactivated.</p>				

Address (hex)		Variable	Description	
<b>41</b>		<b>vWD2</b>	<b>Watchdog (2nd setpoint)</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1 s	Integer	0 ... 150	Professional
<p>Watchdog timer for monitoring data communication.</p> <p>If the timer is set to a value not equal to zero, the Watchdog is active. The variable must be regularly assigned the required time limit, otherwise the "2nd setpoint" is taken over as the setpoint after expiry of the time. After communication has been re-established, the setpoint can be overwritten.</p> <p>If the variable is assigned the value zero, the Watchdog is deactivated.</p>				

Address (hex)		Variable	Description	
<b>42</b>		<b>vSP2</b>	<b>2nd setpoint</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Professional
<p>Setting of the "2nd setpoint". This value should be set to a temperature value at which the process is in a safe state. The setpoint is only activated if certain events occur.</p>				

Address (hex)		Variable	Description	
<b>43</b>		<b>vPMAMode</b>	<b>PMA mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Explore
<p>The so-called PMA mode allows to override the thermostat's temperature controller and directly specify the desired performance (cooling capacity / heating capacity).  The PMA mode must be activated before specifying the capacity (see vPMA, address 0x44).  0: PMA mode not active  1: PMA mode active</p>				

Address (hex)		Variable	Description	
<b>44</b>		<b>vPMA</b>	<b>PMA power specified</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1 %	Integer	-1000 ... 1000	Explore
<p>Power specification as a percentage in PMA mode. Negative values for cooling capacity, positive values for heating capacity. The PMA mode must be activated beforehand (PB command 0x43).</p>				

Address (hex)		Variable	Description	
<b>45</b>		<b>vPMHMode</b>	<b>PMH mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Explore
<p>The so-called PMA mode allows to override the thermostat's temperature controller and to directly specify the desired heating and cooling capacity.  Activate the PMA mode before specifying the power (see VPMH, address 0x46 and vFixCool, address 0x47).  0: PMH mode not active  1: PMA mode active</p>				

Address (hex)		Variable	Description	
<b>46</b>		<b>vPMH</b>	<b>Specifying heating capacity</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1 %	Integer	0 ... 1000	Explore
<p>Heating capacity as a percentage in PMH mode. The PMA mode must be activated beforehand (PB command 0x43).</p>				

Address (hex)		Variable	Description	
<b>47</b>		<b>vFixCool</b>	<b>Specifying cooling capacity</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1 %	Integer	-1 ... 1000	Basic
<p>Cooling capacity specification as a percentage. This function may be used in connection with a power specification in PMH mode (see vPMHMode, address 0x45) or in normal control mode.  The cooling capacity is set to the preset value and the temperature controller automatically adjusts the heating capacity to the desired setpoint if used in normal control mode.</p>				

Address (hex)		Variable	Description	
<b>48</b>		<b>vnPSet</b>	<b>Setpoint pump speed</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1 / min	Integer	0 ... 32000	Basic
<p>Set and query the current pump speed setpoint.  Please note that this function is only available with thermostats with speed-controllable pumps.</p>				

Address (hex)		Variable	Description	
<b>49</b>		<b>vpPSet</b>	<b>Setpoint pump pressure</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1 mbar	Integer	0 ... 32000	Basic
<p>Set and query the current pump pressure setpoint.  Please note that this function is only available with thermostats with speed-controllable pumps or VPC bypass.</p>				

Address (hex)		Variable	Description	
<b>4A</b>		<b>vVPCMode</b>	<b>VPC bypass operating mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
Set and query the operating mode of the VPC bypass. 0: Automatic control of the VPC bypass by the pump pressure setpoint 1: Position setting by automation system				

Address (hex)		Variable	Description	
<b>4B</b>		<b>vVPCPos</b>	<b>VPC bypass target position</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1%	Integer	0 ... 1000	Basic
Specify VPC bypass position or query current setting. Please note that the setting of a VPC bypass position is only possible if the operating mode has first been changed over to position preset (see vVPCMode, Address 0x4A).				

Address (hex)		Variable	Description	
<b>4C</b>		<b>vTKwOut</b>	<b>Cooling water outflow temperature</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Explore
Current measured value of the cooling water outflow temperature.				

Address (hex)		Variable	Description	
<b>4D</b>		<b>vFluidFlow</b>	<b>Thermal fluid volume flow</b>	
Type	LSB	Data type	Value range	E-grade
R	0.1 l/min	Integer	0 ... 10000	Explore
Current measured value of the thermal fluid volume flow. Please note that this measured value is only available if a special volume flow measuring device is provided.				

Address (hex)		Variable	Description	
<b>4E</b>		<b>vFluidFlowSet</b>	<b>Setpoint thermal fluid volume flow</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1 l/min	Integer	0 ... 10000	Explore
Current setpoint of the thermal fluid volume flow. Please note that this function is only available if a special volume flow measuring device is provided. In addition, either a speed-controllable pump or a VPC bypass must be provided in order to allow control.				

Address (hex)		Variable	Description	
<b>4F</b>		<b>vDeltaT</b>	<b>Setpoint delta-T control</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 K	Integer	0 ... 32700	Exclusive
The setpoint is used by the temperature controller. With internal regulation this value is ignored, with process control it is used to limit the temperature difference between the internal and the process temperature. Please note that this setpoint temperature difference can be exceeded depending on the quality of the controller parameterization.				

Address (hex)		Variable	Description	
<b>50</b>		<b>vDeltaTAlarm</b>	<b>Alarm limit delta-T</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 K	Integer	0 ... 32700	Exclusive
Set and query the alarm limit for the difference between the internal and the process temperature. With internal regulation this value is ignored, with process control it is used to monitor the temperature difference between the internal and the process temperature and to trigger a fault if it is exceeded.				

Address (hex)		Variable	Description	
<b>51</b>		<b>vTAlarmHi</b>	<b>Upper alarm limit, internal temperature</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
Set and query upper alarm limit for the internal temperature. If the internal temperature exceeds this value, a fault is generated.				

Address (hex)		Variable	Description	
52		vTIAAlarmLo	<b>Lower alarm limit, internal temperature</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
Set and query lower alarm limit for the internal temperature. If the internal temperature falls below this value, a fault is generated.				

Address (hex)		Variable	Description	
53		vTEAlarmHi	<b>Upper alarm limit, process temperature</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
Set and query upper alarm limit for the process temperature. If the process temperature exceeds this value, a fault is generated.				

Address (hex)		Variable	Description	
54		vTEAlarmLo	<b>Lower alarm limit, process temperature</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.01 °C	Integer	-15111 ... 50000	Basic
Set and query lower alarm limit for the process temperature. If the process temperature falls below this value, a fault is generated.				

Address (hex)		Variable	Description	
55		vOTHeater	<b>Setting of the heating overtemperature protection</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Basic
Query the current setting of the overtemperature protection trigger value of the heating system(s). If the temperature in one of the heating systems rises above this value, a fault is generated.				

Address (hex)		Variable	Description	
56		vOTExpVessel	<b>Setting of the expansion vessel overtemperature protection</b>	
Type	LSB	Data type	Value range	E-grade
R	0.01 °C	Integer	-15111 ... 50000	Basic
Query the current setting of the overtemperature protection trigger value of the expansion vessel. If the temperature in the expansion vessel rises above this value, a fault is generated.				

Address (hex)		Variable	Description	
58		vProgramStart	<b>Starting the temperature control program</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	1 ... 10	Exclusive
Starting the thermoregulation or querying the identifier of the current thermoregulation program. Starts an already entered thermoregulation program.				

Address (hex)		Variable	Description	
59		vRampDuration	<b>Specify ramp duration</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1 s	Integer	-32767 ... 32767	Exclusive
Specify a set point ramp duration or query remaining ramp duration after ramp start. Sets the set point duration of a ramp. The ramp is started by entering a set point using 0x5A (vRampStart). The ramp stops if a value less than 0 is sent while the ramp is running.				

Address (hex)		Variable	Description	
5A		vRampStart	<b>Start ramp</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0,01 °C	Integer	-15111 ... 50000	Exclusive
Enter the end temperature and start the set point ramp or query the end temperature. Specifies the end temperature of a ramp and starts the ramp whose duration was previously defined with command 0x59 (vRampDuration). The final temperature of an active set point ramp can be queried.				



Address (hex)		Variable	Description	
<b>5B</b>		<b>vBlowDownPos</b>	<b>Specifying the blow-down mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0, 2666, 4500, 8266	Basic
Specifying or querying the blow-down mode. 0: Temperature control 2666: Empty 4500: Blow-down 8266: Keep Returns the new target state.				

Address (hex)		Variable	Description	
<b>5C</b>		<b>vMaintenanceDays</b>	<b>Days left until the next maintenance</b>	
Type	LSB	Data type	Value range	E-grade
R	1 d	Integer	-1 ...	Basic
Query the number of days left before the next maintenance message is displayed. The Reminder function has not been activated if the return value is -1.				

Address (hex)		Variable	Description	
<b>5D</b>		<b>vFGasDays</b>	<b>Days left until the next F-gas check</b>	
Type	LSB	Data type	Value range	E-grade
R	1 d	Integer	-1 ...	Basic
Query the number of days left before the next F-gas check is displayed. The Reminder function has not been activated if the return value is -1.				

Address (hex)		Variable	Description	
<b>5E</b>		<b>vServicePackage</b>	<b>Create Service Package</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	-1, 0, 1, 2	Basic
Is passed as Parameter 1 and saves the service package on the USB stick. Return value: -1 (Error), 0 (Initialized), 1 (In Progress), 2 (Successfully Completed).				

Address (hex)		Variable	Description	
<b>5F</b>		<b>vProgramState</b>	<b>Change program status</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 4	Exclusive
Changing the status of a running thermoregulation or querying the current status. Specification: 1: Continue the current thermoregulation program (after a pause) 2: Pause the current thermoregulation program 3: Go to the next segment 4: Terminate the current thermoregulation program Answer: 0: No thermoregulation program active 1: Thermoregulation program runs 2: Thermoregulation program is paused 3: Switched to the next segment 4: Thermoregulation program is finished				

Address (hex)		Variable	Description	
<b>62</b>		<b>vpVPC</b>	<b>VPC Bypass pressure</b>	
Type	LSB	Data type	Value range	E-grade
R	1 mbar	Integer	0 ... 32000	Basic
Absolute pressure measured in the supply line of the VPC bypass. If an external pressure sensor is connected to the VPC bypass, the value measured by that sensor is returned, otherwise the value measured by the pressure sensor in the VPC bypass (at the output of the customer's application) is returned. The return value must be reduced by 1000 mbar if a relative pressure (the difference to the atmospheric pressure) is to be processed.				

Address (hex)		Variable	Description	
69		vTFlowMode	<b>Actual value specification mode for the thermal fluid flow</b>	
Type	LSB	Data type	Value range	E-grade
R / W	--	Bit field	---	Explore
<p>Operating mode to specify the activation, deactivation or querying of the thermal fluid flow.</p> <p>Bit 0:     Operating mode active or inactive</p> <p>          0: Mode not active.</p> <p>          1: Mode active. The volume damper uses the current content of the variable 0x6A (vTFlowVal) to calculate the control variable.</p>				

Address (hex)		Variable	Description	
6A		vTFlowVal	<b>Specification of the thermal fluid flow</b>	
Type	LSB	Data type	Value range	E-grade
R / W	0.1 l/min	Integer	0 ... 10000	Explore
<p>Allows you to specify the actual value for regulating the thermal fluid flow.</p> <p>The command must be used in conjunction with command 0x69 (vTFlowMode), as the specified value will otherwise be ignored. You can verify this with the command 0x4D (vFluidFlow), as it will return the set value if the mode was activated correctly.</p>				

Address (hex)		Variable	Description	
6B		vPumpCtrlMode	<b>Pump control mode</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 3	Basic
<p>Setting and querying the pump control mode.</p> <p>0: Control of pump speed.</p> <p>1: Control of pump pressure.</p> <p>2: Control of the thermal fluid flow.</p> <p>3: Control of the thermal fluid flow rate, but limited to a maximum pressure.</p>				

Address (hex)		Variable	Description	
6C		vPoKoExtMode	<b>External PoKo control</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Explore
<p>Activate or deactivate external control of the PoKo (potential-free contact) or request mode status.</p> <p>0: External control inactive. When the external control is deactivated, the operating mode "Off" is activated.</p> <p>1: External control active. Activation of the external control is only possible if the PoKo is already in the "Off" mode.</p> <p>Note: If a unit has more than one PoKo options (Unistat with Control ONE and additional Com.G@te or POKO / ECS Interface), this command only affects the operating mode of the PoKo on Control ONE.</p>				

Address (hex)		Variable	Description	
6D		vPoKoState	<b>PoKo Status</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Explore
<p>Status of the PoKo (potential-free contact). The setting of the PoKo is only possible if the external control was previously activated with the command 0x6C (vPoKoExtMode).</p> <p>0: PoKo open</p> <p>1: PoKo closed</p> <p>Note: If a unit has more than one PoKo (Unistat with Control ONE and additional Com.G@te or POKO / ECS Interface), this command affects only the status of the PoKo on Control ONE.</p>				

Address (hex)		Variable	Description	
6E		vPowHi	<b>Current power (high bytes)</b>	
Type	LSB	Data type	Value range	E-grade
R	-	Integer	-32767 ... 32767	Explore
<p>Adds the command vPow to query services outside the range -32767 ... 32767.</p> <p>If the device operates in a power range exceeding -32767 ... 32767 W, this command can be used to query the higher-order bytes. To maintain the current, the 2 bytes of the commands vPow and vPowHi must be combined to a signed 4 byte value. Thus vPow forms the lower 2 bytes, vPowHi the high-quality 2 bytes.</p> <p>See also Chapter 9 »PB commands with greater value range and higher accuracy«.</p>				

Address (hex)		Variable	Description	
<b>6F</b>		<b>vAirPurge</b>	<b>Venting</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
<p>Start or stop the venting mode of the thermostat or query the current status. This operating mode is typically required for setting up the system.</p> <p>0: Venting mode not active</p> <p>1: Venting mode active</p>				

Address (hex)		Variable	Description	
<b>70</b>		<b>vDrain</b>	<b>Draining</b>	
Type	LSB	Data type	Value range	E-grade
R / W	1	Integer	0 ... 1	Basic
<p>Start or stop the draining mode of the thermostat or query the current status. This operating mode is typically required for shutting down the system.</p> <p>During draining, all valves in the thermofluid and water circuits of the thermostat are opened.</p> <p>0: Draining mode not active</p> <p>1: Draining mode active</p>				

## 8 PB command examples

### Example 1: Setpoint is to be set to 20 °C

The setpoint for the temperature controller is at Address 0x00.

The value must be transmitted in the unit 0.01 °C. That is to say, a value (in decimal representation) of 2000 must be transmitted. However, this value must be transmitted as a hexadecimal value with 4 characters. In the hexadecimal representation, 2000 corresponds to the value 7D0. To be able to transmit 4 characters, a leading 0 must also be attached. The hexadecimal value must now be converted into a string.

The following string must be sent to the device:

{M0007D0<CR><LF>

In this, <CR> is the symbol for Carriage Return (with the ASCII code 13 or 0x0D) and <LF> is the Line Feed symbol (with the ASCII code 10 or 0x0A).

The thermostat responds (if the value was permissible and has not been limited) with the command: {S0007D0<CR><LF>

### Example 2: Setpoint is to be set to -23.15 °C

A value (in decimal representation) of -2315 must be transmitted. In the hexadecimal representation this corresponds to the value F6F5.

The following string must be sent to the device:

{M00F6F5<CR><LF>

The thermostat responds (if the value was permissible and has not been limited) with the command: {S00F6F5<CR><LF>

### Example 3: Querying the setpoint

If a variable is just to be queried and not set, the character sequence \*\*\*\* is used instead of the value.

Master sends: {M00\*\*\*\*<CR><LF>

Response from thermostat: {S00FFCC<CR><LF>

In this example the thermostat responds with the hexadecimal value FFCC. This corresponds in decimal representation to the value -52. The current setpoint is consequently set at -0.52 °C.

### Example 4: Querying the current internal temperature

Master sends: {M01\*\*\*\*<CR><LF>

Response from thermostat: {S011010<CR><LF>

In this example the thermostat responds with the hexadecimal value 1010. This corresponds in decimal representation to the value 4112. The internal temperature is thus currently 41.12 °C.

### Example 5: Querying the current process temperature (measured at the LEMOSA connector)

Master sends: {M07\*\*\*\*<CR><LF>

Response from thermostat: {S07087F<CR><LF>

The hexadecimal value 087F corresponds to a temperature of 21.75°C.

If the thermostat responds with the command

{S07C504<CR><LF>

this corresponds to a temperature of -151 °C. The value -151 °C means that either no sensor is connected or that the sensor is faulty.

### Example 6: Querying the return temperature

To query the return temperature, the DV E-grade or E-grade Explore must be provided.

The return temperature is at Address 0x02.

Master sends: {M02\*\*\*\*<CR><LF>

Response from thermostat: {S0207E7<CR><LF>

The current return temperature is consequently 20.23°C.

If the necessary E-grade has not been released, the thermostat responds with the command:

{S027FFFF<CR><LF>

The value 7FFF means that either the necessary E-grade is not provided or the queries variable has not been implemented.

**Example 7: Reading off the maximum setpoint**

To make data transmission clearer, in this example, the string to be sent is to be examined again byte by byte.

The command to read off the value of the maximum setpoint limit (Address 0x31) is as follows:

{M31\*\*\*\*<CR><LF>

Each of the characters corresponds to a byte to be transmitted which can be represented differently.

Representation as text:

{ M 3 1 \* \* \* \* <CR> <LF>

Representation of the ASCII code in decimal format:

123 77 51 49 42 42 42 42 13 10

Representation of the ASCII code in hexadecimal format:

7B 4D 33 31 2A 2A 2A 2A 0D 0A

**Example 8: Setting a process temperature of**

Before the mode is switched a suitable process temperature value should be sent to preset to the variable:

Setting a process temperature of 15,12 °C:

Master sends: {M0905E8<CR><LF>

Response from thermostat: {S0905E8<CR><LF>

Switching mode (Watchdog triggers a warning, when no new commands are received):

Master sends: {M190001<CR><LF>

Response from thermostat: {S190001<CR><LF>

From now on a new process temperature value should be continually sent (impossible every second).

Important: Before sending a new command wait for an answer to the previous command.

Setting a process temperature of 15,13 °C:

Master sends: {M0905E9<CR><LF>

Response from thermostat: {S0905E9<CR><LF>

Setting a process temperature of 15,14 °C:

Master sends: {M0905EA<CR><LF>

Response from thermostat: {S0905EA<CR><LF>

Setting a process temperature of 15,15 °C:

Master sends: {M0905EB<CR><LF>

Response from thermostat: {S0905EB<CR><LF>

## 9 PB commands with greater value range and higher accuracy

### INFORMATION

This functionality is only available for devices with Pilot ONE® controller.

The value range of 2-byte values is sometimes insufficient. This applies, for instance, to devices with a temperature range above 300 °C or a power rating above 30 kW. PB commands have therefore been expanded so that larger values can be queried and set too.

The standard format of PB commands always transfers values with 2 bytes (or 4 characters). Adding 2 further bytes (equivalent to 4 additional characters) increases the value range and resolution. In this case, the following features of PB commands change (see Chapter 4 »**Properties of PB Commands**«):

- The length of a command is always 14 characters.
- If an address is contacted, which is not defined or activated (see E-grade), the command returns the same address and the value "7FFFFFFF".

The following properties of some variables change when compared to Chapter 7 »**Special Description of PB commands**«:

- The values of all PB command variables containing a temperature (setpoint temperature, internal temperature, ...) are transferred with 3 decimal places (resolution 0.001 °C instead of 0.01 °C) and a value range of -274000 ... 500000. The measurement range is -200 °C ... 500 °C, non-existent or defective sensors report -274 °C. The additional decimal places provide new information only if the DV-E-grade is activated. The value is otherwise rounded to a hundredths °C and the last digit will always contain the number 0.
- Allows you to record values smaller than -32768 W and greater than 32767 W for all PB command variables containing capacities.
- The values of all PB command variables with volume flow data (thermal fluid flow rate, setpoint thermal fluid flow rate) are transmitted with 3 decimal places (resolution 0.001 l/min instead of 0.1 l/min). The additional decimal place provides new information only if the DV-E-grade is activated. The value is otherwise rounded to a tenth l/min and the last two digits will always contain the number 0.
- Two PB commands vSNRL (0x1B) and vSNRH (0x1C) are required in the standard format for querying the serial number. Both commands provide the complete serial number when the larger value range is used. It is therefore not necessary to send two commands.
- To query the current power, two PB commands are required in the standard format, vPow (0x04) and vPowHi (0x6E), to query values outside the range -32767 ... 32767. Both commands provide the complete power value when the larger value range is used. It is therefore not necessary to send two commands.

A PB command is structured as follows:

{mttvvvvyyy<CR><LF>

{	Start character (ASCII Code 0x7B): Identifies the PB command
m	Sender identifier: Master request "M" (ASCII Code 0x4D) or slave response "S" (ASCII Code 0x53) to identify direction. The automation system (PC, PLC, DCS, ...) is always the master. Its request is designated by an "M". The thermostat is always the slave. The thermostat's response is designated by an "S".
tt	Address (corresponds to a location in a table): 2 characters (1 byte in hex format).
vvvvvyyy	Variable value: 8 characters (4 bytes as hex string) Send the variable value "*****" if you only want to query but not change a variable. The response for a temperature that doesn't exist is -274.000 °C. This corresponds to the string "FFFD1B0".
<CR>	The Carriage Return symbol (ASCII code 0x0D)
<LF>	Das Zeichen Line Feed (ASCII Code 0x0A)

Only the length of the variable value and thus the length of the entire command changes when compared to the standard PB command.

## 9.1 Examples

### Example 9: Setpoint should be set to 20.000 °C (with a higher resolution)

The setpoint for the temperature controller is at Address 0x00.

Specify the value in 0.001 °C for a higher resolution. A value of 20000 (in decimal notation) must be transmitted. This corresponds to the value 00004E20 in hex notation (with 8 characters). The hex value must now be converted into a string.

The following string must be sent to the device:

```
{M0000004E20<CR><LF>
```

In this, <CR> is the symbol for Carriage Return (with the ASCII code 13 or 0x0D) and <LF> is the Line Feed symbol (with the ASCII code 10 or 0x0A).

The thermostat responds (provided the value was permissible and was not limited) with the command:

```
{S0000004E20<CR><LF>
```

### Example 10: Setpoint should be set to -23.150 °C (with a higher resolution)

Transfer a value of -23150 (in decimal notation) for a higher resolution. This corresponds to the value FFFFA592 in hex notation.

The following string must be sent to the device:

```
{M00FFFA592<CR><LF>
```

The thermostat responds (provided the value was permissible and was not limited) with the command:

```
{S00FFFA592<CR><LF>
```

### Example 11: Querying the setpoint (with a higher resolution)

You must use the string \*\*\*\*\* instead of a value if you only want to query but not set a variable.

Master sends:

```
{M00*****<CR><LF>
```

Response from thermostat:

```
{S00 FFFFDF8<CR><LF>
```

In this example the thermostat responds with the hex value FFFFDF8. This corresponds to a decimal value of -520. The current setpoint is therefore set to -0.520 °C.

## 10 PB package command

### INFORMATION

This functionality is only available for devices with Pilot ONE® controller.

Use a PB package command to combine several PB commands into a single command. This considerably speeds up the communication between automation system and thermostat.

Configure the Pilot ONE® controller before using the PB package command. This defines what PB command addresses are to be combined into a PB package command. The package command will then only contain the values of the different PB command addresses to be transferred, not the addresses themselves. The order of the values will correspond to the sequence of the addresses configured on Pilot ONE®. In this order, the PB commands are processed.

### 10.1 Structure

No binary data but ASCII characters are transferred in the communication between automation system (master) and controller (slave). All numerical values are transmitted in hex notation.

The package command is made up as follows:

*[mssillbddd....dddpp<CR>*

[	Start identifier (1 character) The character “[” is defined as start identifier.
m	Communication direction (1 character) The character “M” stands for a master request (from the automation system). The characters “S” stands for a slave response (from the thermostat).
ss	Slave address (2 characters in hex notation) Slave address 1 is used by default. The slave address can be changed in the menu of the Pilot ONE® controller when communicating via the RS232 or RS485 interface of Com.G@tes.
i	Data group identifier (1 character) Letter “B” is defined for the PB package command.
ll	Command length (2 characters in hex notation) The command length corresponds to the number of characters without checksum and without end identifier.
b	Block counter The block counter must be set to 0.
ddd...ddd	PB command address values: (4 characters each) The individual values are transferred as 16-bit numbers. This corresponds to a 4-digit number in hex notation. One character is transferred for each digit. Valid characters are 0-9 and A-Z. The string “****” is also permitted in a master request to tell the controller that this is a read-access-only address. Thus this representation matches the representation of PB command values. A syntactically correct PB package command with a wrong content is indicated when a slave response command only contains the string “EL” or “EB” (including the quotes) instead of the expected values. “EL” indicates that the number of values contained in the master request does not match the configured number. “EB” indicates that an invalid block counter was received. A maximum of 61 PB command addresses can be addressed with one PB package command. The entire command can therefore have a maximum length of 255 characters.
pp	Checksum (2 characters) The ASCII codes of all preceding characters are added in a byte variable (range overflows are ignored). The result equals the checksum and must be entered with 2 characters in hex notation.
<CR>	End identifier <CR> represents the non-printable character Carriage Return (ASCII code 0x0D). This symbol indicates the end of the command.



The value range can also be extended in the package command if it is insufficient. The same characteristics described in section 9 »PB commands with greater value range and higher accuracy« apply.

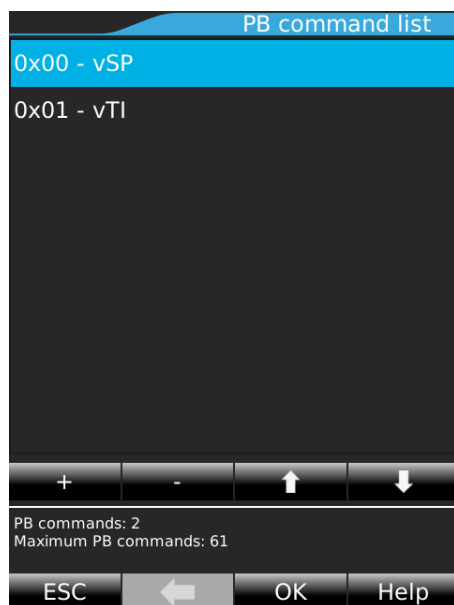
```
[mssillbddddddddd....dddddddddpp<CR>
```

[	Start identifier (1 character) The character “[” is defined as start identifier.
m	Communication direction (1 character) The character “M” stands for a master request (from the automation system). The characters “S” stands for a slave response (from the thermostat).
ss	Slave address (2 characters in hex notation) Slave address 1 is used by default. The slave address can be changed in the menu of the Pilot ONE® controller when communicating via the RS232 or RS485 interface of Com.G@tes.
i	Data group identifier (1 character) Letter “B” is defined for the PB package command.
ll	Command length (2 characters in hex notation) The command length corresponds to the number of characters without checksum and without end identifier.
b	Block counter A maximum of 30 addresses can be addressed with a single command as the system limits the command length to 255 characters. Use the block counters B and C to address more addresses. Use the character “A” for the first block (1 ... 30). Use the character “B” for the second block (31 ... 60). Use the character “C” for the third block (61).
dddddddd...dddddddd	PB command address values: (8 characters each) The individual values are transferred as 32-bit numbers. This corresponds to an 8-digit number in hex notation. One character is transferred for each digit. Valid characters are 0-9 and A-Z. The string “*****” is also permitted in a master request to tell the controller that this is a read-access-only address. Thus this representation matches the representation of PB command values with a wider value range. A syntactically correct PB package command with a wrong content is indicated when a slave response command only contains the string “EL” or “EB” (including the quotes) instead of the expected values. “EL” indicates that the number of values contained in the master request does not match the configured number. “EB” indicates that an invalid block counter was received. A maximum of 30 PB command addresses can be addressed with one PB package command. The entire command can therefore have a maximum length of 251 characters.
pp	Checksum (2 characters) The ASCII codes of all preceding characters are added in a byte variable (range overflows are ignored). The result equals the checksum and must be entered with 2 characters in hex notation.
<CR>	End identifier <CR> represents the non-printable character Carriage Return (ASCII code 0x0D). This symbol indicates the end of the command.

The PB package command is configured in the Pilot ONE® controller. To do so, select the item “PB package” in the main menu “Interfaces”. A default configuration exists without prior user configuration. It can be changed without restrictions. PB command addresses may be added or removed and the sequence of addresses specified.

## 10.4 Examples

The Pilot ONE® controller configuration in the following examples requires each PB package command to address 2 addresses that contain the setpoint (vSP, address 0x00) and the internal temperature (vTI, address 0x01).



This configuration serves as the basis for the following examples.

### Example 1: Querying setpoint and internal temperature

Both configured values need to be queried.

Master sends: [M01B100\*\*\*\*\*2C<CR>

The command is made up as follows:

[ The start identifier.  
M The identifier for a master request.  
01 The slave address was specified as 1.  
B The identifier of the PB package command.  
10 The PB package command length without checksum and terminator is 16 characters, which is 10 characters in hex notation.  
0 The block counter (always 0).  
\*\*\*\* The value of the first configured address (here: vSP, address 0x00, setpoint) is to be queried.  
\*\*\*\* The value of the second configured address (here: vTI, address 0x01, internal temperature) is to be queried.  
2C The checksum calculated by adding the ASCII codes of the preceding characters.  
The PB package command characters  
[ M 0 1 B 1 0 0 \* \* \* \* \* \* \* \*  
match the following values in decimal notation  
91 77 48 49 66 49 48 48 42 42 42 42 42 42 42 42  
The values are added to calculate the checksum:  
91 + 77 + 48 + 49 + 66 + 49 + 48 + 48 + 42 + 42 + 42 + 42 + 42 + 42 + 42 + 42 = 812  
The number 812 equals the number 32C in hex notation. The checksum is 2C as the checksum only uses the last two characters (the lowest byte).  
<CR> The ASCII character Carriage Return.

Response from thermostat: [S01B10007D009F19D<CR>

The first 8 characters of the response match those of the request with the difference that the slave identifier "S" is included instead of the master identifier "M".

In this example, the first value provided by the thermostat is the hex number 07D0, which equals 2000 in decimal notation. Thus the current setpoint is 20.00 °C.

The second value returned is the hex number 09F1, which equals 2545 in decimal notation. Thus the current internal temperature is 25.45 °C.

The checksum of the response is 9D.

**Example 2: Setting the setpoint, querying the internal temperature**

The setpoint should be set to 30.00 °C while the internal temperature is queried.

Master sends: [M01B1000BB8\*\*\*\*70<CR>

Send the hex value 0BB8 to adjust the setpoint to 30.00 °C. Send \*\*\*\* to query the internal temperature.

Response from thermostat: [S01B1000BB809FCC0<CR>

The thermostat returns the setpoint 0BB8 - the new setpoint was therefore applied. The internal temperature value of 09FC is returned, which equals the decimal notation 2556 or 25.56 °C.

**Example 3: Invalid number of values**

The master sends a PB package command with only one value (here "\*\*\*\*"), although 2 values were configured.

Master sends: [M01B0C0\*\*\*\*96<CR>

Response from thermostat: [S01B0C0"EL"C9<CR>

The thermostat returns the error message "EL".

**Example 4: Invalid block counter**

The master sends a PB package command with an invalid block counter. The master sends a block counter value of 1, which the slave responds to with error message "EB".

Master sends: [M01B101\*\*\*\*\*2D<CR>

Response from thermostat: [S01B0C1"EB"C0<CR>

The thermostat returns the error message "EB".

**Example 5: Querying setpoint and internal temperature with a wider value range**

Both configured values need to be queried with a higher resolution.

Master sends: [M01B18A\*\*\*\*\*95<CR>

The command is made up as follows:

[	The start identifier.
M	The identifier for a master request.
01	The slave address was specified as 1.
B	The identifier of the PB package command.
18	The PB package command length without checksum and terminator is 24 characters, which is 18 characters in hex notation.
A	The block counter for the first block.
*****	The value of the first configured address (here: vSP, address 0x00, setpoint) is to be queried.
*****	The value of the second configured address (here: vTI, address 0x01, internal temperature) is to be queried.
95	The checksum
<CR>	The ASCII character Carriage Return.

Response from thermostat: [S01B18A00004E2000003B973C<CR>

The first 8 characters of the response match those of the request with the difference that the slave identifier "S" is included instead of the master identifier "M".

In this example, the first value provided by the thermostat is the hex number 00004E20, which equals 20000 in decimal notation. Thus the current setpoint is 20.000 °C.

The second value returned is the hex number 00003B97, which equals 15255 in decimal notation. Thus the current internal temperature is 15.255 °C.

The checksum of the response is 3C.

**Example 7: Invalid block query with a wider value range**

The master sends a PB package command with two values (here \*\*\*\*\*), however the character "B" is used for the block counter. The second block is not used, as the number of configured PB commands is not greater than the maximum block size for a higher resolution. The thermostat returns the error message "EL".

Master sends: [M01B18B\*\*\*\*\*96<CR>

The command sent is to query variable 31 and 32 of 32 configured addresses. However, the thermostat responds with an error as only 2 addresses were configured.

Response from thermostat: [S01B0CB"EL"DB<CR>

**Example 8: PB package command with a wider value range, if the number of configured PB commands exceeds the maximum block size**

35 PB command variables were combined at the Pilot ONE® controller into one PB package command. This is different from the sample configuration. It exceeds the maximum number of commands that can be combined into one message for a wider range of values (the maximum number is 30 PB command variables). The entire PB package command cannot be processed in a single step. Two PB package commands must be sent instead.

The first PB package command processes the first 30 configured PB command variables. You must use the character "A" for the block counter. The resulting PB package command length without checksum and terminator is 248 characters ( $8 + 30 * 8$ ), which is F8 in hex notation.

The second PB package command processes the following 5 configured PB commands. You must use the character "A" for the block counter. The resulting PB package command length without checksum and terminator is 48 characters ( $8 + 5 * 8$ ), which is 30 in hex notation.

## 11 Modbus

The Modbus protocol allows you to access the PB command described above.

You must also follow the rules described in chapter 5 »**Timing of PB commands**« when communicating with the Modbus protocol.

### INFORMATION

This functionality is only available for devices with Pilot ONE® controller. The Modbus protocol is activated from E-grade Basic, however, the application of various PB command addresses may require activation of additional E-grades (see Chapter 7 »**Special Description of PB commands**«).

### 11.1 General description

Only the general structure of a Modbus command in Modbus TCP is described. Modbus specifications are available on [www.modbus.org](http://www.modbus.org).

### INFORMATION

The Modbus master and the Modbus slave (Thermostat with Pilot ONE® controller) communicate via Ethernet Port 502, which is reserved for Modbus TCP.

The Modbus Slave responds to every syntactically correct Modbus command.

A Modbus command is commonly referred to as “Application Data Unit” (ADU). Each command is made up of 6 blocks (of different lengths):

Modbus Application Protocol Header (MBAP)			
TID	PID	Length	UID
Transaction Identifier	Protocol Identifier	Message Length	Unit Identifier
2 Bytes	2 Bytes	2 Bytes	1 Byte

Protocol Data Unit (PDU)	
FC	Data
Function Code	User Data
1 Byte	n Bytes

The blocks TID, PID, Length and UID are collectively referred to as “Modbus Application Protocol Header” (MBAP) and the blocks FC and Data collectively as “Protocol Data Unit” (PDU).

Transaction Identifier	The <i>transaction identifier</i> (TID) is used to synchronize concurrent requests between server and client.
Protocol Identifier	The <i>protocol identifier</i> (PID) is not used for Modbus TCP but must always be assigned 0x0000.
Message Length	The <i>message length</i> (Length) specifies the number of subsequent bytes in the command (UID + PDU).
Function Code	The <i>function code</i> (FC) specifies the function the Modbus slave is to perform.
Unit Identifier	The <i>unit identifier</i> (UID) has no effect in Modbus TCP, as the slaves are addressed via their IP address. Therefore, this value is always set to 0xFF for Modbus TCP.
User Data	The <i>user data</i> (data) contain the information to be transmitted.

### INFORMATION

Unlike PB commands, where values are converted into strings, binary data are exchanged.

## 11.2 Data types

The following data types are supported:

- Byte (8 Bit)
- Register (16 Bit)
- 32-Bit Integer (32 Bit)

### INFORMATION

Data are transferred as Big Endian according to Modbus specification. That is, the high byte is transmitted first for registers or 32-bit integer values.

## 11.3 Error handling

Two different types of faults must be considered when an error occurs. The client sends no response message if a syntactically incorrect command is sent. This usually leads to a server timeout. An error message is sent back to the server if a syntactically correct command with a wrong content is sent.

Error response:

MBAP Header				PDU	
TID	PID	Length	UID	FC + 0x80	Data (error code)
7 Bytes				1 Byte	1 Byte

MBAP header TID, PID and UID are inherited by the request message. Length = 0x03.

FC + 0x80 0x80 is added to the function code received if an error occurs.

Error code The supported error codes are described in the following table:

Error code	Name	Description
0x01	ILLEGAL FUNCTION	The function code received is not supported by the slave device. A function code may not yet be implemented in the software version of the Pilot ONE® controller if it is described in this manual. Update to a newer software version.
0x02	ILLEGAL DATA ADDRESS	The data address (register address) is not an allowable address.
0x03	ILLEGAL DATA VALUE	The user data length does not match the function code. Maybe there are more or less user data bytes than are required by the corresponding function.
0x04	SLAVE DEVICE FAILURE	A configuration on the slave device prevents the use of this functionality.

## 11.4 Specific description of function codes

The function codes (FC) supported in thermostats with Pilot ONE® controller are listed below.

Two types of Modbus function codes are supported:

- Public Function Codes
- User-Defined Function Codes

### INFORMATION

Of the **Public Function Codes** according to Modbus specifications, only the functions 0x03 (Read Holding Registers) and 0x06 (Write Single Register) are supported.

These are based on the PB commands (see Chapter 7 »**Special Description of PB commands**») and transmitted as a 2-byte value (register).

**INFORMATION**

The following provisions apply to **User-Defined Function Codes** based on PB commands: Modbus commands are based on PB commands with a wider range of values. This means that values must always be transmitted as 4 bytes (see Chapter 9 »**PB commands with greater value range and higher accuracy**« and Chapter 10.2 »**Structure with greater value range and higher accuracy**«). Furthermore, all provisions apply to PB commands with a wider range of values.

**INFORMATION**

Only exemplary transaction identifier (TID) values are used in the examples for each function code. However, it can be chosen arbitrarily and has no effect on the execution of the function.

### 11.4.1 Public Funktionscodes

#### 11.4.1.1 Read Holding Registers (FC 0x03)

The function code 0x03 is used to query several consecutive registers. The addresses of the registers can be found in chapter 7 (»**Special Description of PB commands**«).

PDU of request:

FC	Data	
0x03	Start address	Number
1 Byte	2 Byte	2 Bytes

PDU of response:

FC	Data	
0x03	Number of consecutive bytes	Values
1 Byte	1 Byte	N * 2 Bytes

Error handling:

Name	Description
ILLEGAL DATA ADDRESS (0x02)	<ul style="list-style-type: none"> <li>The start address is greater than the greatest PB command address.</li> <li>The sum of the start address + number of PB addresses to be read is greater than the greatest PB command address.</li> </ul>
ILLEGAL DATA VALUE (0x03)	<ul style="list-style-type: none"> <li>The length of the query is not correct.</li> <li>The number of PB addresses to be read is incorrect. It is limited by the maximum number of available PB addresses.</li> </ul>

#### Example 1: Querying the first 3 PB command addresses

The first 3 PB command addresses (setpoint temperature VSP (0x00), internal temperature vTI (0x01), return temperature VTR (0x02)) should be queried.

The client sends the following request to the Modbus slave:

**00 01 00 00 00 06 FF 03 00 00 00 03**

The command is made up as follows:

00 01	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 06	The message length is 6 bytes. Of these, 1 byte is used for the unit identifier, 1 byte for the function code, 2 bytes for the start address and 2 bytes for the number of PB addresses to be read.
FF	Unit identifier (UID, always 0xFF)
03	Function code 0x03 for Reading Holding Registers
00 00	Start address (vSP 0x00)
00 03	3 consecutive PB addresses should be read

The Modbus slave returns the following response:

**00 01 00 00 00 09 FF 03 06 08 98 01 2C FE 0C**

The response is made up as follows:

00 01 TID (inherited from query)  
 00 00 PID (always 0x0000)  
 00 09 The message length is 9 bytes. Of these, 1 byte is used for the unit identifier, 1 byte for the function code, 1 byte for the number of consecutive bytes and 6 bytes (3 \* 2 bytes) are used for the values of the PB variable.  
 FF Unit identifier (UID, always 0xFF)  
 03 Function code 0x03 for Reading Holding Registers  
 06 The number of following bytes (3 read addresses with 2 bytes each)  
 08 98 The value of the setpoint temperature is 0x0898. This equals 2200 in decimal notation and corresponds to a temperature of 22.00 °C.  
 01 2C The value of the internal temperature is 0x012C. This equals 300 in decimal notation and corresponds to a temperature of 3.00 °C.  
 FE 0C The value of the return temperature is 0xFE70. This equals -500 in decimal notation and corresponds to a temperature of -5 °C.

#### 11.4.1.2 Write Single Holding Register (FC 0x06)

The function code 0x06 is used for writing a single PB address. The addresses of the registers can be found in chapter 7 (»Special Description of PB commands«).

PDU of request:

FC	Data	
0x06	PB address	Value
1 Byte	2 Byte	2 Bytes

PDU of response:

FC	Data	
0x06	PB address	Value
1 Byte	2 Byte	2 Bytes

Error handling:

Name	Description
ILLEGAL DATA ADDRESS (0x02)	▪ The PB address is greater than the greatest PB command address.
ILLEGAL DATA VALUE (0x03)	▪ The length of the query is not correct.

#### Example 2: Changing setpoint temperature (positive value)

The value of the setpoint temperature (VSP 0x00) should be changed to a positive value. This value is within the setpoint limits.

The client sends the following request to the Modbus slave:

**00 02 00 00 00 06 FF 06 00 00 05 DC**

The command is made up as follows:

00 02 TID (any number that can be used to assign a response to a query)  
 00 00 PID (always 0x0000)  
 00 06 The message length is 6 bytes. Of these, 1 byte is used for the unit identifier, 1 byte for the function code, 2 bytes for the PB address and 2 bytes are used for the value to be written.  
 FF Unit identifier (UID, always 0xFF)  
 06 Function code 0x06 is used for Writing Single Holding Registers  
 00 00 PB address (vSP 0x00)  
 05 DC The setpoint temperature is to be set to 15.00 °C. The value 1500 equals 0x05DC in hex notation.

The Modbus slave returns the following response:

**00 02 00 00 00 06 FF 06 00 00 05 DC**

In this case, the response is an echo of the query. This shows that the setpoint temperature was correctly set to 15.00 °C.



**Example 3: Changing setpoint temperature (negative value)**

The value of the setpoint temperature (VSP 0x00) should be changed to a negative value. This value is within the setpoint limits.

The client sends the following request to the Modbus slave:

**00 03 00 00 00 06 FF 06 00 00 F8 30**

The command is made up as follows:

00 03	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 06	The message length is 6 bytes. Of these, 1 byte is used for the unit identifier, 1 byte for the function code, 2 bytes for the PB address and 2 bytes are used for the value to be written.
FF	Unit identifier (UID, always 0xFF)
06	Function code 0x06 is used for Writing Single Holding Registers
00 00	PB address (vSP 0x00)
F8 30	The setpoint temperature is to be set to -20.00 °C. The value -2000 equals 0xF830 in hex notation.

The Modbus slave returns the following response:

**00 03 00 00 00 06 FF 06 00 00 F8 30**

In this case, the response is an echo of the query. This shows that the setpoint temperature was correctly set to -20.00 °C.

**Example 4: Changing setpoint temperature (negative value is limited)**

The value of the setpoint temperature (VSP 0x00) should be changed to a negative value. This value is outside of the setpoint limits.

The minimum setpoint limit is set to -30.00 °C at the Pilot ONE.

The client sends the following request to the Modbus slave:

**00 04 00 00 00 06 FF 06 00 00 F2 54**

The command is made up as follows:

00 04	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 06	The message length is 6 bytes. Of these, 1 byte is used for the unit identifier, 1 byte for the function code, 2 bytes for the PB address and 2 bytes are used for the value to be written.
FF	Unit identifier (UID, always 0xFF)
06	Function code 0x06 is used for Writing Single Holding Registers
00 00	PB address (vSP 0x00)
F2 54	The setpoint temperature is to be set to -35.00 °C. The value -3500 equals 0xF254 in hex notation.

The Modbus slave returns the following response:

**00 04 00 00 00 06 FF 06 00 00 F4 58**

The response is made up as follows:

00 04	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 06	The message length is 6 bytes. Of these, 1 byte is used for the unit identifier, 1 byte for the function code, 2 bytes for the PB address and 2 bytes are used for the inherited value.
FF	Unit identifier (UID, always 0xFF)
06	Function code 0x06 is used for Writing Single Holding Registers
00 00	PB address (vSP 0x00)
F4 58	The value of the setpoint temperature is 0xF448. This equals -3000 in decimal notation and corresponds to a temperature of -30.00 °C. It follows that the setpoint temperature was limited to the lower setpoint limit.

## 11.4.2 User-Defined Functioncodes

### 11.4.2.1 Communication test (FC 0x41)

This function is used to test the communication with the Modbus slave. The response returns the request unchanged.

PDU of request:

FC	Data
0x41	-
1 Byte	0 Bytes

PDU of response:

FC	Data
0x41	-
1 Byte	0 Bytes

#### Example 1: Communication test

The client sends the following request to the Modbus slave to test the communication:

**00 01 00 00 00 02 FF 41**

The command is made up as follows:

00 01 Transaction identifier (TID, any number that can be used to assign a response to a query)  
 00 00 Protocol identifier (PID, always 0x0000)  
 00 02 The message length (Length, length of the subsequent bytes) is 2 bytes. 1 byte is used for the unit identifier and 1 byte is used for the function code. User data are not used in a communication test.  
 FF Unit identifier (UID, always 0xFF)  
 41 Function code (FC) 0x41 for the communication test

In this case, the response of the Modbus slave has always the same structure:

**00 01 00 00 00 02 FF 41**

### 11.4.2.2 Query PB variable (FC 0x42)

The function code 0x42 is used to query a single PB command address. This address can be found in the table in chapter 7 (»Special Description of PB commands«).

PDU of request:

FC	Data
0x42	PB address
1 Byte	1 Byte

PDU of response:

FC	Data	
0x42	PB address	Value
1 Byte	1 Byte	4 Bytes

#### Example 2: Querying the internal temperature (vTI)

The internal temperature is to be queried. According to the table, the PB variable "vTI" has the address 0x01.

The client sends the following request to the Modbus slave:

**00 02 00 00 00 03 FF 42 01**

The command is made up as follows:

00 02	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes. 1 byte is used for the unit identifier, 1 byte for the function code and 1 byte is used for the PB address.
FF	Unit identifier (UID, always 0xFF)
42	Function code 0x42 to query the PB variable
01	Use address 0x01 to query PB variable vTI.

The Modbus slave returns the following response:

**00 02 00 00 00 07 FF 42 01 00 00 5B A0**

The response is made up as follows:

00 02	TID (inherited from query)
00 00	PID (always 0x0000)
00 07	The message length is 7 bytes. 1 byte is used for the unit identifier, 1 byte for the function code, 1 byte for the PB address and 4 bytes are used for the value of the PB variable.
FF	Unit identifier (UID, always 0xFF)
42	Function code 0x42 to query the PB variable
01	PB variable "vTI" (0x01)
00 00 5B A0	The value of the internal temperature is 0x00005BA0. This equals 23456 in decimal notation and corresponds to a temperature of 23.456 °C.

Note: Data are always transmitted with increased resolution in Modbus TCP (see Chapter 9 »PB commands with greater value range and higher accuracy«).

#### Example 3: Querying a PB variable that does not exist

An error response is returned if a non-existent PB variable is queried. In this case, a PB variable with the address 0xFA (250) is to be queried.

The client sends the following request to the Modbus slave:

**00 03 00 00 00 03 FF 42 FA**

The command is made up as follows:

00 03	TID (inherited from query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes.
FF	Unit identifier (UID, always 0xFF)
42	Function code 0x42 to query the PB variable
FA	Use the address 0xFA to query the (non-existent) PB variable

The Modbus slave returns the following error response:

**00 03 00 00 00 03 FF C2 03**

The response is made up as follows:

00 03	TID (inherited from query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes.
FF	Unit identifier (UID, always 0xFF)
C2	Function code (0x42) + 0x80 (see Chapter 11.3 »Error handling«)
03	Error code 0x03 ILLEGAL DATA VALUE

#### 11.4.2.3 Changing and querying a PB variable (FC 0x43)

Use this function to change a PB command variable. You must specify the PB address in the user data according to Chapter 7 (»Special Description of PB commands«). You must also specify the value that is to be added. The Modbus slave's response contains the current value after the change. As an alternative to FC 0x42, you can also query values without changing them. To do so, specify 0x7FFFFFFF in the user data (Data) for the variable value to be transmitted.

PDU of request:

FC	Data	
0x43	PB address	Value
1 Byte	1 Byte	4 Bytes

PDU of response:

FC	Data	
0x43	PB address	Value
1 Byte	1 Byte	4 Bytes

#### Example 4: Starting the temperature control process

The PB variable “vTmpActive” for starting the temperature control process is located at address 0x14.

The client sends the following request to the Modbus slave:

**00 04 00 00 00 07 FF 43 14 00 00 00 01**

The command is made up as follows:

- 00 04 TID (any number that can be used to assign a response to a query)
- 00 00 PID (always 0x0000)
- 00 07 The message length is 7 bytes. 1 byte is used for the unit identifier, 1 byte for the function code, 1 byte for the PB address and 4 bytes are used for the PB variable value that is to be written.
- FF Unit identifier (UID, always 0xFF)
- 43 Function code 0x43 for changing and querying a PB variable
- 14 Assign address 0x14 to the PB variable “vTmpActive”.
- 00 00 00 01 Send the value 0x00000001 to start the temperature control process.

The Modbus slave returns the following response:

**00 04 00 00 00 07 FF 43 14 00 00 00 01**

The command is made up as follows:

- 00 04 TID (inherited from query)
- 00 00 PID (always 0x0000)
- 00 07 7 more bytes will follow in this message.
- FF Unit identifier (UID, always 0xFF)
- 43 Function code 0x43 for changing and querying a PB variable
- 14 PB variable “vTmpActive” (address 0x14)
- 00 00 00 01 State of temperature control process. 1 means temperature control process is active.

#### 11.4.2.4 Querying a PB package command (FC 0x44)

Use this function to query multiple variables with a single package command. First you have to configure the package as described in Chapter 10.3 »Configuration«.

The query’s user data only contain the number of configured PB variables. An error response is returned if it doesn’t agree with the number of PB variables configured in the Pilot ONE® controller.

The response’s user data also contain the values of the queried PB variables.

PDU of request:

FC	Data
0x44	Number addresses
1 Byte	1 Byte

PDU of response:

FC	Data			
0x44	Number Addresses	Value 1	...	Value n
1 Byte	1 Byte	4 Bytes	...	4 Bytes

#### Example 5: Querying configured addresses

For this example, the PB package command was configured as follows:

- vSP (0x00) Setpoint
- vTI (0x01) Internal temperature

The client sends the following request to the Modbus slave:

**00 05 00 00 00 03 FF 44 02**

The command is made up as follows:

00 05	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes (UID + FC + Data).
FF	UID (always 0xFF)
44	Function code 0x44 to query a PB package command
02	Number of configured variables in PB package command.

The Modbus slave returns the following response:

**00 05 00 00 00 0B FF 44 02 00 00 61 A8 FF FF EC 78**

The command is made up as follows:

00 05	TID (inherited from query)
00 00	PID (always 0x0000)
00 0B	The message length is 11 (0x000B) bytes. 1 byte is used for the unit identifier (UID), 1 byte for the function code, 1 byte for the number of PB addresses and 2 x 4 bytes are used for the current value of the PB variables.
FF	UID (always 0xFF)
44	Function code 0x44 to query a PB package command.
00 00 61 A8	Setpoint: 0x000061A8 is 25000 in decimal notation and therefore equals a temperature of 25.000 °C if a higher resolution is used.
FF FF EC 78	Internal temperature: 0xFFFFEC78 is -5000 in decimal notation and therefore equals a temperature of -5.000 °C if a higher resolution is used.

#### Example 6: Query does not match the configuration

For this example, the PB package command was configured as follows:

- vSP (0x00) Setpoint
- vTI (0x01) Internal temperature

The client sends the following request to the Modbus slave:

**00 06 00 00 00 03 FF 44 03**

The command is made up as follows:

00 06	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes (UID + FC + Data)
FF	UID (always 0xFF)
44	Function code 0x44 to query a PB package command.
03	This value does not match the number of PB variables configured in the PB package command (2 variables were configured but 3 variables were queried).

The Modbus slave returns the following error response:

**00 06 00 00 00 03 FF C4 03**

The command is made up as follows:

00 06	TID (inherited from query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes (UID + FC + Data)
FF	UID (always 0xFF)
C4	Function code 0x44 + 0x80 (see Error handling)
03	Error code 0x03 ILLEGAL DATA VALUE

#### Example 7: Configured PB package command does not contain PB variables

An error response is returned upon a query if the configured PB package command does not contain PB variables.

The client sends the following request to the Modbus slave:

**00 07 00 00 00 03 FF 44 02**

The command is made up as follows:

00 07	TID (any number that can be used to assign a response to a query)
00 00	PID (always 0x0000)
00 03	The message length is 3 bytes (UID + FC + Data)
FF	UID (always 0xFF)
44	Function code 0x44 to query a PB package command.
02	It is assumed that two PB variables are configured in the PB package command.

The Modbus slave returns the following error response:

**00 07 00 00 00 03 FF C4 04**

The command is made up as follows:

00 07 TID (inherited from query)  
00 00 PID (always 0x0000)  
00 03 The message length is 3 bytes (UID + FC + Data)  
FF UID (always 0xFF)  
C4 Function code 0x44 + 0x80 (see Chapter 11.3 »Error handling«)  
04 Error code 0x04 "SLAVE DEVICE FAILURE"

#### 11.4.2.5 Changing and querying a PB package command (FC 0x45)

Use this feature to modify multiple variables with a single command and to query the state after a change. First you have to configure a package as described in Chapter 10.3 »Configuration«.

The query's user data contain the number of configured PB variables. An error response is returned if it doesn't agree with the number of PB variables configured in the Pilot ONE® controller. Send the value 0x7FFFFFFF to those variables that should be read only, when you only want to modify specific variables with the PB package command while others should remain unchanged.

The response has the same structure as the query.

PDU of request:

FC	Data			
0x45	Number Ad- dresses	Value 1	...	Value n
1 Byte	1 Byte	4 Bytes	...	4 Bytes

PDU of response:

FC	Data			
0x45	Number Ad- dresses	Value 1	...	Value n
1 Byte	1 Byte	4 Bytes	...	4 Bytes

#### Example 8: Modifying the setpoint temperature, querying the remaining variables

For this example, the PB package command was configured as follows:

- vSP (0x00) Setpoint
- vTI (0x01) Internal temperature
- vTmpActive (0x14) State of temperature control process

The client sends the following request to the Modbus slave:

**00 08 00 00 00 0F FF 45 03 00 00 53 FC 7F FF FF FF 7F FF FF FF**

The command is made up as follows:

00 08 TID (any number that can be used to assign a response to a query)  
00 00 PID (always 0x0000)  
00 0F The message length is 15 (0x000F) bytes. 1 byte is used for the unit identifier (UID), 1 byte for the function code, 1 byte for the number of PB addresses and 3 x 4 bytes are used for the PB variables' values to be written.  
FF UID (always 0xFF)  
45 Function code 0x45 to change and query the PB package command  
03 Number of configured variables in PB package command.  
00 00 53 FC Set setpoint temperature to 21.500 °C (21500 equals 0x000053FC)  
7F FF FF FF Internal temperature can be queried only  
7F FF FF FF The state of the temperature should not be changed, only queried

The Modbus slave returns the following response:

**00 08 00 00 00 0F FF 45 03 00 00 53 FC 00 00 61 40 00 00 00 01**

The command is made up as follows:

00 08 TID (inherited from query)  
00 00 PID (always 0x0000)  
00 0F 15 bytes (0x000F) follow  
FF UID (always 0xFF)  
45 Function code 0x45 to change and query the PB package command  
03 Number of configured variables in PB package command  
00 00 53 FC Setpoint: 0x000053FC equals 21.500 °C  
00 00 61 40 Internal temperature: 0x00006140 equals 24.896 °C  
00 00 00 01 Temperature control process active

## 12 OPC-UA

### 12.1 What is OPC UA?

OPC UA is a protocol used to provide device data to a higher-level controller. It has the advantage that its data is self-explanatory. That is, the OPC server sends a description of its data and thus enables automatic interpretation. For example, when a temperature is transmitted, it not only transmits a numerical value but also its unit, a brief description of the temperature (e.g. set point), the upper and lower limit of the value and its data type. This eliminates otherwise necessary conversions from m °C to °C in PB commands.

### 12.2 Description of function

#### 12.2.1 What can the Huber OPC server do?

You can use the Huber OPC server to query and change various values of the thermostat. It is based on the PB commands displayed in the OPC tree.

The OPC server is activated in the Pilot ONE® Menu, in the “Interfaces” category. The OPC server requires the E-grade “OPC UA”.

#### **INFORMATION**

All PB commands are enabled for communication with the E-grade “OPC-UA”.

#### 12.2.2 Number of clients

A maximum of 3 connections to the server are possible in Pilot ONE®. A connection is rejected and a client-based error message appears if you try to connect more clients to the Huber OPC server.

#### 12.2.3 Speed / Polling interval

The Huber OPC server data is updated once per second. It is thus sufficient for the client to query its data once every second.

#### 12.2.4 Subscriptions

In addition to permanent reading (polling) of OPC variables, OPC-UA offers the possibility of subscriptions. An OPC client can add several variables to a subscription. These variables are then called monitored items. The OPC server then automatically monitors these monitored items and informs the client of changes, e.g. if the value of the variable has changed. The Huber OPC server supports one subscription with a maximum of 100 monitored items per connected client.

#### **INFORMATION**

For certain OPC clients (e.g. UaExpert®) a subscription for ModelChange events is automatically created. If further OPC variables are then to be observed in a new subscription, this fails with the error BadTooManySubscriptions (since only one subscription per client is supported). If possible, this option should be deactivated. The OPC tree of the Huber OPC server is static and does not change at runtime.

## 12.3 Prerequisites

### 12.3.1 Network

Generally, OPC only communicates via Ethernet. You can connect a temperature control unit either by integrating it into an existing network or by connecting it to a client via a direct point-to-point connection. Each type of connection requires different Ethernet interface settings. The default host name is set to Pilot ONE <Serial Number>. The Huber OPC server responds to the OPC port 4840. Please refer to the Pilot ONE® Manual for further details on configuring the Ethernet interface.

### 12.3.2 Certificates

#### NOTE

##### Communication with Pilot ONE is unencrypted

##### DAMAGE TO THE TEMPERATURE CONTROL UNIT AND THE APPLICATION

- Communication with Pilot ONE must be unencrypted.
- Communication encryption can be deactivated on Pilot ONE. This mode is **not** recommended for production environments. It is only used for debugging connections or for special applications that make encryption superfluous!  
Unencrypted communication allows data to be read and/or manipulated in the network! Further measures must be taken to ensure data integrity! e.g. 1:1 connection within a specially secured installation space or by means of other IT technical processes.

The connection between the Huber OPC server and the client is encrypted to ensure a secure communication. Thus, third parties cannot eavesdrop or manipulate the data. The encryption is based on certificates, which are exchanged between both participants and the participants use certificates to introduce each other. Huber OPC server certificate are valid for 10 years. The user is warned one month before the certificate expires.

## 12.4 Setup mode

### 12.4.1 Pilot ONE® Configuration

Pilot ONE® must be configured before you can establish a connection with the temperature control unit. The following settings must be made:

- Time
- Ethernet
- Huber OPC Server

#### INFORMATION

Changing certain settings on the temperature control unit may require the generation of a new certificate. A newly generated certificate must be newly exchanged between temperature control unit and client.



#### 12.4.1.1 Setting the time

The time and the time zone are very important parts of the SSL certificates as the period of validity of the SSL certificates is limited. If the time specifications are incorrect, the connection will fail because the certificates may not yet be valid or are already invalid in relation to the time settings.

### PROCEDURE

- Go to the “Categories Menu”.
- Tap on the category “System Settings”.
- Tap on the category “Clock, Calendar”.
- Make the changes needed and confirm them by tapping “OK”.

### INFORMATION

Please also note that the time settings on the client system must also be correct.

#### 12.4.1.2 Ethernet set-up

IP address, host name, domain name and DNS servers must be configured before certificates can be exchanged. These settings are applied to the certificates. Certificates must be recreated if settings are changed later. A newly generated certificate must be newly exchanged with the client.

### PROCEDURE

- Go to the “Categories Menu”.
- Tap on the category “Interfaces”.
- Tap on the category “Ethernet”.
- Make the changes needed and confirm each by tapping “OK”.

## 12.5 Setting a Huber OPC Server

### 12.5.1 Start/stop

The Huber OPC server must be started from the Pilot ONE to enable a connection between the client and the temperature control unit. The icon is greyed out and the Huber OPC server cannot be started if the “OPC UA” E-grade is not active.

### PROCEDURE

- Go to the “Categories Menu”.
- Tap on the category “Interfaces”.
- Tap on the category “OPC UA”.
- Tap on the entry “Start/Stop”.

You have the following options, if the Huber OPC server is not started:

- **Start (once)** is only valid until the next restart of the temperature control unit. The Huber OPC server is automatically stopped after each restart of the temperature control unit.
- **Enable (always)** starts the Huber OPC server and saves the settings. The Huber OPC server is automatically started after each restart of the temperature control unit.

You have the following options once the Huber OPC server was started:

- **Stop** disables the Huber OPC server if the previous mode was “Start (once)”. The Huber OPC server is stopped and the settings are saved if the mode was “Enable (always)”. Thus, the Huber OPC server is not started automatically after restarting the temperature control unit.

## 12.5.2 Encryption

### NOTE

#### Communication with Pilot ONE is unencrypted

#### DAMAGE TO THE TEMPERATURE CONTROL UNIT AND THE APPLICATION

- Communication with Pilot ONE must be unencrypted.
- Communication encryption can be deactivated on Pilot ONE. This mode is **not** recommended for production environments. It is only used for debugging connections or for special applications that make encryption superfluous!  
Unencrypted communication allows data to be read and/or manipulated in the network! Further measures must be taken to ensure data integrity! e.g. 1:1 connection within a specially secured installation space or by means of other IT technical processes.

The connection between Huber OPC Server and client is encrypted by default. However, it is also possible to deactivate the encryption. Yet, this setting can only be changed if the Huber OPC server is not yet active.

## PROCEDURE

- Go to the "Categories Menu".
- Tap on the category "Interfaces".
- Tap on the category "OPC UA".
- Tap on the entry "Encryption".

## 12.5.3 Configuration

This menu item explains what is needed to operate the Huber OPC server and what settings are required on the Pilot ONE.

## 12.5.4 Exchanging certificates

Certificates must be exchanged between the Pilot ONE® or the Huber OPC server and the client to ensure a secure communication. These certificates are exchanged between client and Pilot ONE by USB stick as these certificates are essential for a secure communication and because a manipulation of such certificates would prevent a secure communication.

The Pilot ONE® certificate must be imported into the client and the client certificate must be imported in Pilot ONE®.

Pilot ONE® stores its certificates in the directory "server\_certs", which is located in the root directory of the USB stick. The client certificates must be located in the root directory. The creation of client certificates depends on the manufacturer and is described in its instructions.

### INFORMATION

New certificates are generated if the Ethernet interface settings are changed. These must then be redistributed to the clients.

- **Import (from a USB stick)**  
Client certificates can be imported from the USB stick. These must be located in the root directory of the USB stick. The certificates that need to be imported can be selected individually.
- **Export (to a USB stick)**  
The server certificate is exported to the USB stick. It can then be found in the directory "server\_certs". The file name corresponds to the host name.
- **Remove (internally)**  
Use this item to remove invalid or no longer needed certificates.
- **Display (internally)**  
The imported client certificates are displayed.
- **Generate new (overwrite)**  
The server certificate of the temperature control unit is regenerated. The new certificate is valid for another 10 years. The public part of the certificate must be redistributed to the clients.



# Inspired by **temperature** designed for you

Peter Huber Kältemaschinenbau AG  
Werner-von-Siemens-Str. 1  
77656 Offenburg / Germany

Telefon +49 (0)781 9603-0  
Telefax +49 (0)781 57211

[info@huber-online.com](mailto:info@huber-online.com)  
[www.huber-online.com](http://www.huber-online.com)

Technischer Service: +49 (0)781 9603-244

-125 °C ... +425 °C

**huber**