# Data Communication

Version 1.3/04.08
PETER HUBER KÄLTEMASCHINENBAU GMBH
WERNER-VON-SIEMENS-STRASSE 1
D-77656 OFFENBURG-ELGERSWEIER
PHONE: (+49) 7 81 / 96 03 - 0 TELEFAX (+49) 7 81 / 5 72 11
INTERNET http://www.huber-online.com E-MAIL Info@huber-online.com

# Data Communication

## Content

# Data Communication

## 1. General Rules

### 1. Requirements on the reliability of the application

The controller is equipped with a bidirectional RS232 interface and an RS485 semi-duplex interface. The hardware can be switched in the menu item „Digital Interface“, using either RS232 or RS485.

During the initial operation, the following sequence should be observed for the digital interface:
(1) All participants are switched off.
(2) Establish the connections.
(3) Switch on the devices.
(4) Start data transmission.

**Reliability and Hardware**

Evaluate the requirements on the reliability of your application:

Hardware:
You can chose between RS232 and RS485.
If several thermostats are operated, a real bus operation with RS485 is unavoidable. The RS485 can also be used to control one thermostat which increases the reliability of data transmission.

General notes on the reliability of the data transmission:
In general, it is difficult to state a value which quantifies the error rate during data transmission. However, it can be generalized that in an office environment an RS232 data connection offers sufficient safety, in particular at cable lengths of 3 m. On the other hand, it is quite well possible that the safety drops rapidly without being noticed because the shielding is not okay or has been connected in the wrong way, for example. By reducing the transmission rate, the safety is increased automatically. In an industrial environment, the RS485 is used in general.
An error in transmission should occur no more frequent than once a week. In such cases in particular, the control software should be so robust that no visible limitations occur.

# Data Communication

## 1. Generale Rules



**Security and Software**

Software protocol:

(1) Point-to-point commands (PP commands):

These commands are intended for the data exchange of precisely two participants.
They have an easy-to-understand structure; the transmission is not secured by check sums, for this reason the communication software is very simple as well. A control by echo is always possible.

(2) Bus commands (bus protocols):

These commands are suitable for use by more than two participants. Special rules are agreed upon, according to which the commands are set up. A check sum and, in particular, a receiver address have been implemented, which requires, of course, that a clear address standard for all bus participants exists.
The LAI commands belong to this class.
The bus protocols can also be used for an application with one participant only! The higher reliability in the bus protocols result from the protection via check sum and the „good-natured" behavior of the slaves in faulty transmission.

(3) General properties of the protocols

PP commands: All default properties of a command are checked. Only if all characters match the command agreed upon, a parameter is changed and a reply is sent, if and when required. Of course, not all errors can be recognized. If the thermostat receives SP@ a21<CR><LF>, for example, a syntax error has occurred and the thermostat remains passive. If, however, SP@ 221<CR><LF> is received instead of SP@ 321<CR><LF>, the error remains unnoticed and the thermostat reacts; the syntax is correct, but the content is wrong. The echo evaluation can cause a correction quickly.
LAI commands: The rules are more extensive; the data consistency in the commands is always checked. The slave will not respond if a violation occurs.

**Master / Slave principle**

Master / slave principle

All protocols apply this principle. The thermostat is always the slave, which consequently means that the thermostat becomes active only, if a request exists. The master transmits to the slave and always receives an answer (echo), when the command received has been received in perfect condition. No slave may transmit unsolicited.
*Response behavior of a thermostat*

The master / slave principle already indicates that no slave may send a request unsolicited. This behavior is further intensified: If a command is used in an impermissible manner, no parameters will be transferred and no response either!

*Examples*

sP( +00a10<CR><LF> There are three errors here.
SP? +02000<LF><CR> There are three errors here as well.

If a hexadecimal figure contains foreign characters or if small letters are used for A ... F, the error condition is fulfilled, and the thermostat does nothing.

# Data Communication

## 1. General Rules

4. Software test

**Software test**

Notes on the software test
If you have written or have to evaluate a program, please check the behavior in extreme situations.
Please check how the system behaves when the thermostat or the PC or both are affected by a short-time mains failure.
In this context, please observe that the parameters in the thermostat have to be set up again after a mains failure.
A short-time data failure should be tested as well.

Transmission format

| | |
|---|---|
| Baud rate | 1200, 2400, 4800, **9600,** 19200 |
| Process | asynchronous |
| Start bit | 1 |
| Data bits | 8 |
| Parity | none |
| Stop bit | 1 |
| Handshake | No |

Time behavior (timing)

The software protocols shall have to be structured in such a way that very simple timing rules can be applied:
The data flow within a command should not be interrupted. Pauses of more than 100 ms between the characters of a command will cause the receiver to abort the command in the process of reception.
The protocols have been set up in such a way that an „echo“ can always be received. If the echo has been received, the next command can be transmitted immediately.
The typical response time is below 300 ms. If no echo is used, it is recommended to wait for 1 s between two commands.

# 2. General Rules - Syntax

## 1. Preparation of software protocols – number formats

**Notation**

A differentiation is made between upper and lower case.Please keep to these requirements strictly.

**Number formats**

Four number formats are used:

(1) Decimal number format Z1 (for PP commands)
The other formats are used in LAI commands:
(2) Hexadecimal number format Z2 for check sums
(3) Hexadecimal number format Z3 for temperatures with a 16-bit resolution (2 bytes, thus 4 ASCII characters)

Number format Z1

In PP commands which contain a figure as a parameter, this number format is used:
Snnnnn S stands for the prefix key, thus „+“ or „-“.
The five-digit number is symbolized by „nnnnn“ (one numeral 0 to 9 in ASCII format each).

The number format is used in transmitting as well as in receiving commands.
As a receiver, the controller is tolerant within certain limits with respect to the number format, i.e. the prefix key „+“ can be left out and preceding zeros are not required either. Experimental values have revealed, however, that the user software is more robust, if the format of a command is observed precisely.

Number formats Z2, Z3

Numbers are rendered as hexadecimal characters in complement to two.
The ASCII characters „0“ to „9“ and „A“ to „F“ are used in the transmission of numbers.

Each byte is represented by two ASCII characters,
where apply:

| Number format | Number of bytes | Number of ASCII characters | 100 (decimal) |
|---|---|---|---|
| Z2 | 1 | 2 | 64 |
| Z3 | 2 | 4 | 0064 |

The decimal 100 corresponds to hexadecimal characters 64h.
This number is represented by the ASCII characters „6“ and „4“.

# Data Communication

## 2. General Rules - Syntax

2. Preparation of software protocols – temperature formats
3. General parameter properties

**Temperature formats**

Temperature formats Z1, Z3 and Z4

All formats are based on the Celsius temperature scale.
Where: 0° Celsius = 0.

Temperature in Z1 format

LSB is 0.01K.
Thus, the writable temperature range extends to +999.99° Celsius.
4° Celsius is represented as +00400.
-4° Celsius is represented as -00400.

Temperature in Z3 format

LSB is 0.01K.
Thus, the writable temperature range extends to 7FFF, thus 327.67° Celsius.
4° Celsius is represented as 0190.
-4° Celsius is represented as FE70.

**General parameter properties**

An attempt is made to set up easy and generally applicable rules for the handling of parameters, wherever possible. Then the command description indicates only where the general properties are not adhered to.

Temporal applicability of the parameters

The controller of the thermostat contains a memory which ensures that the setting of the thermostat (its „parameterization“) is available again after the mains has been switched on.
This memory is referred to as being permanent.

Caution! The permanent memory cannot be written any number of times!
It is limited to 100.000 writing cycles due to the component construction.

In addition, there is a volatile memory, the main memory which can be written as often as required.
In manual operation it is not possible to reprogram the permanent memory that the limits are reached.
A „badly“ written control program is more likely to achieve this.
For this reason, the operating variables are only written into the volatile memory. Thus, for some important variables the possibility was found to cause a permanent storage of the parameter.

Limiting overranging:

If a parameter setting is required which is outside the valid range, the parameter is limited to the maximum or minimum value!

Breaking-in assistance for software development: Every invalid command received from the controller is represented by a small dot at the bottom right-hand side in the „Normal“ or „Double“ display. It will be deleted only after the display has been refreshed.

# 2. General Rules - Syntax

4. General structure of a PP command

**PP command**

The structure of the PP commands shall be demonstrated by the command for the setpoint value.
SP# Snnnnn<CR><LF>
„SP“ represents the command string as transmitted from the master PC to the thermostat. Directly followed by #. # is a character which indicates how the command is to be executed. Snnnnn is a number in Z1 format which describes the parameter. Snnnnn is always preceded by a blank character. <CR><LF> represents the termination. <CR><LF> is not preceded by a blank character.

Permanent storage

In some commands, such as SP, it is possible to explicitly cause permanent storage. The command description points out to this fact. If # = & is written, the parameter will stored volatile and permanently. After the mains is switched on again, this value is available again, provided no subsequent manual input has been made.
**Caution!** The permanent memory cannot be written any number of times! It is limited to 100.000 writing cycles due to the component construction.

Significance of „#“

# = ? the setpoint value is to be retrieved; the echo is requested.
Master: SP?<CR><LF>
Slave: SP+02000<CR><LF>

# = @ the setpoint value is overwritten and the echo is requested.
Master: SP@+02100 <CR><LF>
Slave: SP+02100 <CR><LF>

# = & the setpoint value is overwritten volatile and permanently; no echo is requested (this option is available for some parameters only).
Master: SP&+02100 <CR><LF>
Slave: SP+02100 <CR><LF>

#=! the setpoint is only overwritten, no echo is erquested.
(if # is not available ‘!’ is assumed).
Master: SP! +02200 <CR><LF>
(alternative: SP+02200 <CR><LF>)
Slave: the value 22°C is taken over however there is no echo (not recommendable variant).

The PP commands can be used only for the exchange of data between two partners using the number format 1 (Z1).

# Data Communication

## 3. PP Commands

1 Setpoint values
2 Setpoint value limitation and alarm values

**PP commands - examples**

Examples for the presentation of the PP commands:

Example 1: A command which permits one value to be set and retrieved only.

| Master | Slave |
|---|---|
| SP# Z1<CR><LF> | SP Z1<CR><LF> |

Example 2: A command which makes sense only in conjunction with a query:

| Master | Slave |
|---|---|
| TI?<CR><LF> | TI Z1<CR><LF> |

Example 3: A command, in which the variable is not temperature, but a condition. For ease in handling, the number format Z1 is used although only one of the two conditions has to be set. These two conditions are set by the numerical values TRUE and FALSE. The numerical values have to be adhered to precisely.

Where:
TRUE = +00001 (condition 'internal control')
FALSE = +00000 (condition 'external control' (process control / cascade control))

| Master | Slave |
|---|---|
| TM# Z1<CR><LF> | TM Z1<CR><LF> |

**Setpoint values**

Basic properties which deviate from the „General parameter properties".
Setpoint values

Setpoint default (temperature)

| Master (&: permanently possible) | Slave |
|---|---|
| SP# Z1<CR><LF> | SP Z1<CR><LF> |

Range of input: Permanent minimum and maximum setpoint value.

Priority: A potential setpoint default of the analogue interface is higher!

Second setpoint value – is frequently used to enter a safe condition in case of process failures.

| Master (&: permanently possible) | Slave |
|---|---|
| SP2# Z1<CR><LF> | SP2 Z1<CR><LF> |

Range of input: Permanent minimum and maximum setpoint value.

**Setpoint value limitation**

**Alarm values**

Setpoint value limitations and alarm values
Minimum setpoint value limitation (temperature)
**Caution! These variables can be saved permanently only.**

| Master (&: permanently possible) | Slave |
|---|---|
| LL# Z1<CR><LF> | LL Z1<CR><LF> |

Range of input: Minimum device temperature and permanent setpoint value.

Maximum setpoint value limitation (temperature)

| Master (&: permanently possible) | Slave |
|---|---|
| LH# Z1<CR><LF> | LH Z1<CR><LF> |

Range of input: Maximum device temperature and permanent setpoint value.

# Data Communication

## 3. PP Commands

3. Alarm temperatures
4. Actual values
5. Control mode
6. KeyLock Mode

**Alarm temperatures**

Alarm temperatures:

**Caution! These variables can be saved permanently only.**
higher Alarm temperature

| Master | Slave |
|---|---|
| AA# Z1<CR><LF> | AA Z1<CR><LF> |

Range of input: device temperature range

lower Alarm temperature

| Master | Slave |
|---|---|
| AI# Z1<CR><LF> | AI Z1<CR><LF> |

Range of input: device temperature range.

**Actual value**

Actual value
Temperatures query. Bath and/or inlet temperature in the Z1 format

| Master | Slave |
|---|---|
| TI?<CR><LF> | TI Z1<CR><LF> |

Prozess-Temperatur (des externen Pt100) im Z1 Format

| Master | Slave |
|---|---|
| TE?<CR><LF> | TE Z1<CR><LF> |

**Control mode**

Adjust the control mode.

| Master (&: permanently possible) | Slave |
|---|---|
| TM# Z1<CR><LF> | TM Z1<CR><LF> |

The mode ‚external control mode' is being activated with TRUE.
The mode ‚internal control mode' is being activated with FALSE.

**Thermo-regulation**

Activate/deactivate thermoregulation

| Master | Slave |
|---|---|
| CA# Z1<CR><LF> | CA Z1<CR><LF> |

The thermoregulation is activated with TRUE
The thermoregulation is deactivated with FALSE.

**KeyLock Mode**

Lock the keyboard

| Master | Slave |
|---|---|
| KL# Z1<CR><LF> | KL Z1<CR><LF> |

The keyboard is locked with TRUE.
The keyboard is unlocked with FALSE.

# Data Communication

## 3. PP Commands

7. Watchdog
8. Potential-free contact Poko

**Watchdog**

The watchdog
The watchdog is a piece of safety equipment which permits the communication between thermostats and PCs to be monitored. In order to activate the watchdog, a WDx command has to be transmitted to the thermostat together with a time value. Within this period of time, the command has to be refreshed as otherwise the watchdog will carry out its action. If the WD1 command is active, the watchdog switches the thermoregulation off, and an error message is displayed. If the WD2 command is active, the watchdog replaces the setpoint value with the second setpoint value, which is then used to heat/cool the thermostat. A warning message is displayed.

Put the watchdog in the mode WD1.

| Master | Slave |
|---|---|
| WD1# Z1<CR><LF> | WD1 Z1<CR><LF> |

Put the watchdog in the mode WD2.

| Master | Slave |
|---|---|
| WD2# Z1<CR><LF> | WD2 Z1<CR><LF> |

Z1 renders the watchdog time in seconds. The maximum value is 150 s.
At Z1 = 0, the watchdog is switched off.

**Poko**

Handling potential-free contact (POKO)
In order to switch the potential-free contact by means of the RS interface, the PKRS mode has to be activated with the PKRS command first. If the PKRS is deactivated, the PK commands will be ignored. However, the PK? command will always return the current POKO status. When the PKRS mode is switched on, the POKO remains in the current condition until the first PK command is received; when the system is switched off, the condition of the controller determines the POKO status.

PKRS-determine mode

| Master | Slave |
|---|---|
| PKRS# Z1<CR><LF> | PKRS Z1<CR><LF> |

TRUE sets PKRS on active / FALSE sets PKRS on inactive.

| Master | Slave |
|---|---|
| PK# Z1<CR><LF> | PK Z1<CR><LF> |

Es gilt: TRUE designates the activated POKO (coil is ernergised).
FALSE designates the inactive POKO (coil is not ernergised).

# Data Communication

## 4. LAI Command Group

1. General principles

**Principles**

The LAI command group

A number of bus-compatible commands are available under the protocol designation of LAI.

The „General guidelines“ of the software protocols are applicable. In addition, there are the following special features:

LAI command structure

A LAI command is structured as follows:
**„[mssilld...dpp\r„**.

with:

| | | |
|---|---|---|
| **[** | start character 5Bh | 1 byte |
| **m** | transmitter identifier M (4Dh) for master or S (53h) for slave | 1 byte |
| **ss** | slave address 01 ... 99 | 2 bytes |
| **i** | identifier of the data group | 1 byte |
| **ll** | length of the data field | 2 bytes |
| **d...d** | data group | 0 ...50 bytes |
| **pp** | check sum | 2 bytes |
| **\r** | rogue indicator CR 0Dh | 1 byte |

**Definition**

The transmitter identifier indicates the direction of the data traffic. All characters in front of the check sum are referred to as data field.
The data group are the characters after the seventh byte up to the check sum.
The actual data are contained in the data group. The significance of the data is determined by the identifier and the transmitter identifier.
Below the commands will be referred to according to the identifier of the data group.

In order to increase the data safety, a check sum is transmitted. The check sum is the 1 byte sum of all hex values from the start character to the last character in front of the check sum.

Example: The Master sends : **„[M01V07C6\r„**

| | ASCII | Hex | Meaning |
|---|---|---|---|
| 1.Byte | [ | 5Bh | Start signal |
| 2.Byte | **M** | 4Dh | transmitter identification M = Master |
| 3.Byte | **0** | 30h | Slaveaddress |
| 4.Byte | **1** | 31h | Slaveaddress |
| 5.Byte | **V** | 56h | Identifier data group |
| 6.Byte | **0** | 30h | Length of the data field (07 Byte) |
| 7.Byte | **7** | 37h | Length of the data field (07 Byte) |
| 8.Byte | **C** | 43h | Check sum |
| 9.Byte | **6** | 36h | Check sum |
| 10.Byte | **\r** | 0Dh | End-character CR |

# 4. LAI Command Group

2. Queries and responses

In order to be able to query variables without changes, it is possible to set the „*“ character instead of a variable in the master command. The receiver, thus the thermostat, will not change the variable at this position. In this case, **all positions** which the variable takes, have to be rendered with the „*“ character.
In the examples, the thermostat is always addressed with the identifier (device address) 01.

**´V´ Verify**

Provided to check the presence of a slave.

**Master query:**
**„[M01V07C6\r“** The master queries whether the slave 01 is connected to the bus.
**Slave answer:**
**„[S01V0EMINI CCAD\r“** Slave 01 is connected to the bus, and the device is a MINI CC. The slave command has the „MINI CC“ data group, which is 7 bytes long. These 7 bytes plus the 7 bytes in front of the data group produce a data field length of 14 bytes = 0Eh byte.

**´G´ General**

This command transmits the most important temperatures and the status information. A modified setpoint value is not saved to the permanent memory, i.e. this value is lost when the mains is switched off.

**Master query**

**Master query: „[M01G0Dsattttpp\r“**

s = Control mode:
- ´C´ = Circulation, switch on the circulation.
- ´E´ = Switch on the external control mode.
- ´I´ = Switch on the internal control mode.
- ‘O’ = Off, standby mode.
- ´*´ = Make no change of the current condition.

a = Cancelling the alarm:
- ´0´ = No Alarm cancellation.
- ‘1´ = A possible alarm is being canceled.
- ´*´ = Make no change of the current condition.

tttt = Setpoint value in the Temperature format Z3.
- ´****´ = No change of the setpoint value.

pp = Check sum format Z2

\r End-character CR.

# 4. LAI Command Group

2. Queries and responses

**Slave-response**

**Slave response: „[S01G15sattttiiiieeeepp\r„**

s = Control mode:
- ´C´ = Circulation, switch on the circulation.
- ´E´ = Switch on the external control mode.
- ´I´ = Switch on the internal control mode.
- ´O´ = Off, standby mode.

a = Alarm status:
- ´0´ = No Alarm.
- ´≠≠ 0´ = A number other than 0 means alarm.

tttt = Setpoint value in the temperature format Z3.

iiii = Internal actual value in the temperature format Z3.

eeee = External actual value in the temperature format Z3.

pp = Check sum in format Z2.

\r End-character CR.

# 4. LAI Command Group

2. Queries and responses

**´L´ Limit**

‚L' Limit This command transfers the setpoint value limits;
**Master query**: **„[M01L0Fllllhhhhpp\r„**

llll = Low-Limit, lower setpoint limit in temperature format Z3.
**** = No change of the lower setpoint limit.
hhhh = High-Limit, upper setpoint limit in temperature format Z3.
**** = No change of the upper setpoint limit.
pp = Check sum in the Format Z2.
\r End-character CR.

**Slave response**: **„[S01L17llllhhhhuuuuoooopp\r„**

llll = Low-Limit, lower setpoint limit in temperature format Z3.
hhhh = High-Limit, upper setpoint limit in temperature format Z3.
uuuu = Lower working-range limit. This limit is specific to the device and cannot be modified. The lower setpoint limit cannot be below the lower working range limit.
oooo = Upper working range limit. This limit is specific to the device and cannot be modified. The upper setpoint limit cannot be above the upper working range limit.
pp = Check sum in format Z2.
\r End-character CR.

# 4. LAI Command Group

2. Queries and responses

**´I´**
**Ident**

´I´ Ident This command modifies the slave address.

**Caution! The address is stored in the permanent memory.**

The memory is component-related limited to 10 000 write cycles. When programming attention must be paid that this parameter is not overwritten too often.

**Master query**: **„[M01I09nnpp\r„**

nn = New Slave address ´01´...´99´. ** no modification of the Slave address.

pp = Check sum in format Z2

\r End-character CR.

**Slave response**: **„[S01I09nnpp\r„**

nn = New Slave address. This address will be valid after the answer is sent. If ** was stated as the new slave address, the previous slave address will be returned.

pp = Check sum in format Z2

\r End-character CR

# 4. LAI Command Group

2. Queries and responses

**´A´ Alarm-grenzen**

‚A' Alarm limit
This command transfers the alarm limits (ref. to menu alarm configuration);

**Master query: „[M01A0Fllllhhhhpp\r„**

llll = Low-Alarm, lower alarm value in temperature format Z3.
**** = No modification of the lower alarm value.
hhhh = High-Alarm, upper alarm value in temperature format Z3
**** = No modification of the upper alarm value.
pp = Check sum in Format Z2.
\r End-character CR.

**Slave response: „[S01A0Fllllhhhhpp\r„**

llll = Low-Limit, lower setpoint limit in temperature format Z3.
hhhh = High-Limit, upper setpoint limit in temperature format Z3.

pp = Check sum in format Z2.
\r End-character 0CR.