Table of contents:

EbV Elektronikbau- und Vertriebs- GmbH, Heisterner Weg 8-12, 57299 Burbach

1 Integration of the heatapp! and heatcon! systems into home automation

heatapp! and heatcon! systems can be integrated into home automation from version 2.2.42898 onwards. To do this, we use the Modbus or MQTT interfaces / with Homeassistant.

To enable operation via home automation, the corresponding interface must be activated in the System management menu. The following chapters describe which data points are readable and writable.
A new menu has been implemented in the system for remote access.

1.1 Available values for reading/writing

You can read/write the following values using Modbus/MQTT/Homeassistant:

Outdoor temperature 1	Read
Outdoor temperature 2	Read
Room temperature 1 - n	Read
Room setpoint temperature 1 – n	Read / Write
Heating circuit flow temperature 1 – n	Read
Heating circuit setpoint temperature 1 – n	Read
Hot water actual temperature	Read
Hot water setpoint temperature	Read / Write
Energy generator flow temperature	Read
Energy generator return temperature	Read

2 Modbus in heatapp! and heatcon! systems

When integrated into a Modbus system, heatapp! / heatcon! systems become Modbus servers (slaves). Data points are queried using the TCP IP protocol.
Modbus TCP/IP is a modern extension of the Modbus protocol for communication via Ethernet networks. It uses the TCP protocol to transfer data packets via IP networks, including LANs, WANs and even the Internet. This variant enables easier integration into IT networks.

2.1 Activating the Modbus connection

Select the Remote menu. Activate the connection to the Modbus server and enter the network data of your Modbus master.

The network data of the Modbus master determines the network from which the heatapp! / heatcon! devices expect messages and send responses. For Modbus communication, the communication partner must be located in the defined network.

NOTE:
Use the Modbus default port 502!

2.2 Reading and writing data points

The data points are accessed in the input and holding registers. The address range is between 30001 and 49999. All input registers are read-only. Holding registers are read-write.

2.3 List of data points

The following data points are used in the heatapp! / heatcon! system for reading and writing values:

Data value	First Modbus address	Read/Write	Format	Function code
Outdoor temperature 1	30001	Read Only	Sensor	Input Register
Outdoor temperature 2	30002	Read Only	Sensor	Input register
HK1 room temperature	31102	Read Only	Sensor	Input register
HK1 room setpoint temperature	41101	Read Write	Temperature	Holding register
HK2 room temperature	31202	Read Only	Sensor	Input Register
HK2 room setpoint temperature	41201	Read Write	Temperature	Holding register
HK3 room temperature	31302	Read Only	Sensor	Input Register
HK3 room setpoint temperature	41301	Read Write	Temperature	Holding register
HK4 room temperature	31402	Read Only	Sensor	Input Register
HK4 room setpoint temperature	41401	Read Write	Temperature	Holding register
HK5 room temperature	31502	Read Only	Sensor	Input Register
HK5 room setpoint temperature	41501	Read Write	Temperature	Holding register
HK6 room temperature	31602	Read Only	Sensor	Input Register
HK6 room setpoint temperature	41601	Read Write	Temperature	Holding register
HK7 room temperature	31702	Read Only	Sensor	Input Register
HK7 room setpoint temperature	41701	Read Write	Temperature	Holding register
HK8 room temperature	31802	Read Only	Sensor	Input Register
HK8 room setpoint temperature	41801	Read Write	Temperature	Holding register
HK9 room temperature	31902	Read Only	Sensor	Input Register
HK9 room setpoint temperature	41901	Read Write	Temperature	Holding register
HK10 room temperature	32002	Read Only	Sensor	Input Register
HK10 room setpoint temperature	42001	Read Write	Temperature	Holding register
HK11 room temperature	32102	Read Only	Sensor	Input register
HK11 room setpoint temperature	42101	Read Write	Temperature	Holding register
HK12 room temperature	32202	Read Only	Sensor	Input Register
HK12 room setpoint temperature	42201	Read Write	Temperature	Holding register
HK13 room temperature	32302	Read Only	Sensor	Holding register
HK13 room setpoint temperature	42301	Read Write	Temperature	Holding register
HK14 room temperature	32402	Read Only	Sensor	Input Register
HK14 room setpoint temperature	42401	Read Write	Temperature	Holding register
HK15 room temperature	32502	Read Only	Sensor	Input Register
HK15 room setpoint temperature	42501	Read Write	Temperature	Holding register
WW actual temperature	32602	Read Only	Temperature	Input Register
WW setpoint temperature day	42601	Read Write	Temperature	Holding register
WW setpoint temperature night	42602	Read Write	Temperature	Holding register
EEZ 1 flow temperature	33104	Read Only	Sensor	Input register
EEZ 1 return temperature	33105	Read Only	Sensor	Input register
HK1 flow setpoint temperature	31104	Read Only	Temperature	Input register
HK1 flow temperature	31105	Read Only	Sensor	Input register
HK2 flow setpoint temperature	31204	Read Only	Temperature	Input register

Data value	First Modbus address	Read/Write	Format	Function code
HK2 flow temperature	31205	Read Only	Sensor	Input register
HK3 flow setpoint temperature	31304	Read Only	Temperature	Input register
HK3 flow temperature	31305	Read Only	Sensor	Input register
HK4 flow temperature setpoint	31404	Read Only	Temperature	Input register
HK4 flow temperature	31405	Read Only	Sensor	Input register
HK5 flow setpoint temperature	31504	Read Only	Temperature	Input register
HK5 flow temperature	31505	Read Only	Sensor	Input register
HK6 flow setpoint temperature	31604	Read Only	Temperature	Input register
HK6 flow temperature	31605	Read Only	Sensor	Input register
HK7 flow setpoint temperature	31704	Read Only	Temperature	Input register
HK7 flow temperature	31705	Read Only	Sensor	Input register
HK8 flow setpoint temperature	31804	Read Only	Temperature	Input register
HK8 flow temperature	31805	Read Only	Sensor	Input register
HK9 flow setpoint temperature	31904	Read Only	Temperature	Input register
HK9 flow temperature	31905	Read Only	Sensor	Input register
HK10 flow setpoint temperature	32004	Read Only	Temperature	Input register
HK10 flow temperature	32005	Read Only	Sensor	Input register
HK11 flow setpoint temperature	32104	Read Only	Temperature	Input register
HK11 flow temperature	32105	Read Only	Sensor	Input register
HK12 flow setpoint temperature	32204	Read Only	Temperature	Input register
HK12 flow temperature	32205	Read Only	Sensor	Input register
HK13 flow setpoint temperature	32304	Read Only	Temperature	Input register
HK13 flow temperature	32305	Read Only	Sensor	Input register
HK14 flow setpoint temperature	32404	Read Only	Temperature	Input register
HK14 flow temperature	32405	Read Only	Sensor	Input register
HK15 flow setpoint temperature	32504	Read Only	Temperature	Input register
HK15 flow temperature	32505	Read Only	Sensor	Input register

The following outputs can be displayed:

Data format	Data type	Value	Function
Sensor	Signed short	-32768	No sensor
		-32767	Sensor interruption
		-32766	Sensor short circuit
		-32758	Sensor digital status OFF
		-32757	Sensor digital status ON
		-500 … 5000	Temperature value -50°C … 500.0°C
Temperature	Signed short	-32767	Sensor interruption as temperature value
		-32766	Sensor short circuit as temperature value
		-32765	--.-°C
		-32764	Sensor not yet ready as temperature value
		-32763	Type error
		-32762	Modem automatic (SHORT CIRCUIT)
		-32761	Modem automatic (INTERRUPTION)
		-32760	Modem automatic (MEASURED VALUE KTY 65.0°C)
		-32758	Digital value 0
		-32757	Digital value 1
		-32756	Special value Initiate shutdown
		-32755	Shutdown detection (setpoint goes)
		-32754	Shutdown detection (setpoint coming)
		-32753	Validity limit for sensor values
		-32752 … 32767	Temperature value -3275.2 … 3276.7°C
Locking	(Not listed anywhere)		

Data format	Data type	Value	Function
Warning message	(Not listed anywhere)		
Digital	Unsigned short	0	OFF
		1	ON
		2 … 65535	undefined

2.4 Invalid queries

If a value is requested that is not enabled for communication via Modbus, the value 65535 is returned.

3 MQTT in heatapp! and heatcon! systems

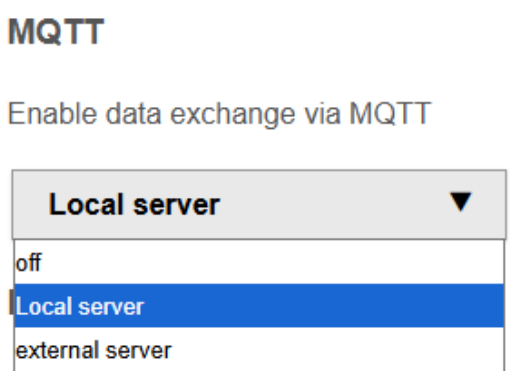
The abbreviation MQTT originally stands for Message Queuing Telemetry Transport. It is a lightweight communication protocol that was developed specifically for wireless sensor networks. For this reason, it is frequently used in automation technology, industrial communication and, last but not least, in the Internet of Things (IoT). Devices that were originally decoupled from each other can thus communicate directly with each other.

The MQTT architecture consists of a server, also known as a broker, one or more clients and a network. The broker is the central intermediary that enables communication between the clients. It is responsible for receiving, storing and sending messages to the clients.
The clients are the end devices that communicate with the broker. They can send messages to the broker and receive messages from the broker. Such a message is also called an MQTT message.
The network provides the connection between the broker and the clients. This can be either a local network or the internet.

There are various ways to enable data exchange via MQTT:

Allow data exchange via	• **Off** • Local server • External server
Encryption	• **No** • Yes
Authentication method	• **None** • Password • Certificate
MQTT topic prefix	• **None** • To be entered individually
Home Assistant	• **No** • Yes

3.1 Allow MQTT data exchange



First, select how you want to allow data exchange via MQTT.

3.1.1 MQTT off

In disabled mode, the device does not allow external access. Ports 1883 and 8883 are closed. No messages are sent.

3.1.2 MQTT via local server

When selecting the local server, data exchange from the local network is permitted. The heatcon! listens for messages from the address 0.0.0.0.
When encryption is disabled, port 1883 is used.
When encryption is enabled, port 8883 is used.

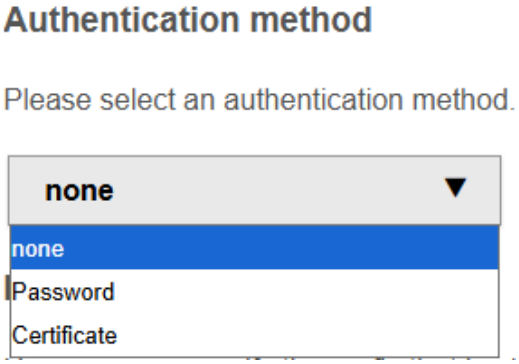
3.1.3 MQTT via external server (bridge)

First, enter the address and port of your external server. Data exchange takes place exclusively with the defined broker, which generates the topics.

3.2 Authentication method

Choose between the following authentication methods

- None
- Password
- Certificate



NOTE:
Ensure that the port is entered correctly in the broker.
Encrypted data exchange = port 8883
Unencrypted data exchange = port 1883

3.2.1 No authentication

If "NONE" authentication is selected as the access method, no further settings are necessary. The MQTT master can listen to the specified topic and write the corresponding values.

3.2.2 Password

For the "PASSWORD" access method, a user and password must be created and saved in heatcon! EC.

3.2.3 Certificate

NOTE:
The option to authenticate using a certificate is only available if encryption is enabled. Without encryption, the only options available are no authentication or authentication using a password.

For authentication using a certificate, the certificates are generated by heatcon! EC in the local network. These certificates are stored in heatcon! EC. By clicking on the "Export as file" button, all certificates are exported individually as txt files and saved on your PC. Make sure you note the storage location so that you can then integrate the files into your MQTT broker.

NOTE:
Importing and saving externally generated certificates in PEM format is only possible if you operate the MQTT master using an external server (bridge). When using a local server, only certificates generated by heatcon! EC can be used.

3.2.4 Validity of certificates

The certificates are valid for 10 years. The expiry date is displayed above each certificate.

3.3 MQTT topic prefix

Enter the prefix that will precede all topics here. The prefix allows you to structure your MQTT topics and thus manage them more easily.

3.3.1 Structure of MQTT topics

Different values are sent depending on the configuration of the heatapp! / heatcon! system.

MQTT specification for heatapp!/heatcon! systems
Topic structure
MQTT topics of the heatapp! / heatcon! systems begin with a prefix that can be customised in the system configuration. The prefix can be anything but must be unique for each heatapp! / heatcon! system connected to the same broker.

Special topic used for an active query of all values, e.g. immediately after setting up the client.

Topic	Description	Published by
/<prefix>/update_all	Requests that all available values be sent	Client

The topics are structured as follows:

Topic	Description	Published by
/<prefix>/<component>/temperature/state	Current temperature	heatapp! / heatcon!
/<prefix>/<component>/setpoint/state	Current setpoint	heatapp! / heatcon!

Topic	Description	Published by
/<prefix>/<component>/setpoint/set	New setpoint (write)	Client
/<prefix>/<component>/setpoint/mode/	Setpoint operating mode, normally "Auto"	heatapp! / heatcon!

Available topics

For better readability, the suffixes "state/set/mode" have been omitted in this table.

Topic	Description	Read / write
/outdoor/sensor_1/temperature	Current temperature of outdoor sensor 1	Read
/outdoor/sensor_2/temperature	Current temperature outdoor sensor 2	Read
/zone_<n>/temperature	Current temperature zone n (zone n: rooms 1-24)	Read
/zone_<n>/setpoint	Setpoint for zone n (zone n: room 1-24)	Read / write
/water_heater/temperature	Current hot water temperature	Read
/water_heater/day/setpoint	Day setpoint for hot water	Read / write
/water_heater/night/setpoint	Night hot water setpoint	Read / write
/boiler/forward_flow/temperature	Boiler flow temperature	Read
/boiler/return_flow/temperature	Boiler return flow temperature	Read
/circuit_<n>/forward_flow/temperature	Current flow temperature for heating circuits n (n: 1-15)	Read
/circuit_<n>/forward_flow/setpoint	Flow setpoint temperature for heating circuits n (n: 1-15)	Read

4 Homeassistant in heatapp! / heatcon! systems

The heatapp! / heatcon! system can be added to a Homeassistant system using MQTT.
Activate support for Homeassistant in the Remote menu.

4.1 Homeassistant setup example

4.2 Preparing heatapp!/heatcon! for Homeassistant

- Select remote menue
- Activate MQTT
- Activate Home Assistant

4.3 Preparing Homeassistant

The MQTT integration is added in Homeassistant. Depending on the desired setting, either the official Mosquitto MQTT broker add-on is used or the MQTT broker connection details are entered manually.

Once all settings have been made, the heatapp! base / heatcon! EC is automatically detected and displayed in the Homeassistant overview.