akribis

USER GUIDE

AQT560

Vaisala · Authorized Partner

Distributed by AKRIBIS

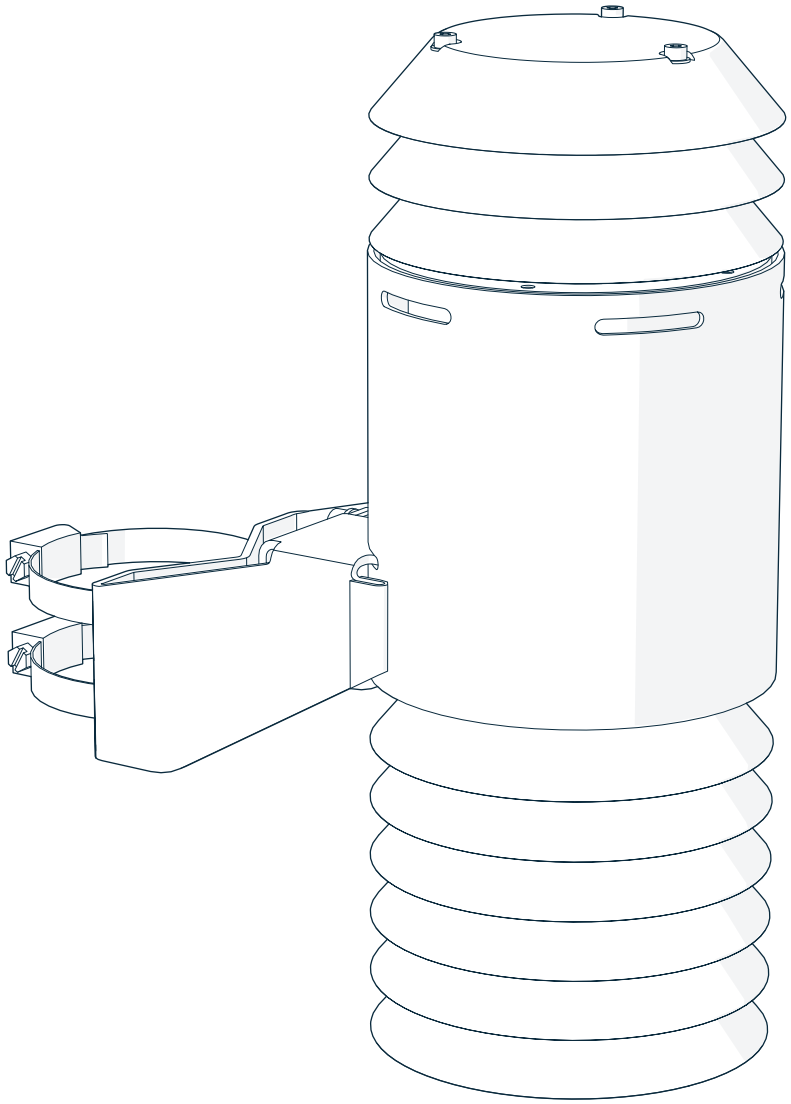


M213026EN-A

Configuration Guide

Vaisala Air Quality Transmitter

AQT560



VAISALA

AKRIBIS · www.akribis.info · contacto@akribis.info

PUBLISHED BY

Vaisala Oyj
Vanha Nurmijärventie 21, FI-01670 Vantaa, Finland
P.O. Box 26, FI-00421 Helsinki, Finland
+358 9 8949 1
www.vaisala.com
docs.vaisala.com

© Vaisala 2024

No part of this document may be reproduced, published, or publicly displayed in any form or by any means, electronic or mechanical (including photocopying), nor may its contents be modified, translated, adapted, sold, or disclosed to a third party without prior written permission of the copyright holder. Translated documents and translated portions of multilingual documents are based on the original English versions. In ambiguous cases, the English versions are applicable, not the translations.

The contents of this document are subject to change without prior notice.

Local rules and regulations applicable to the products and services may vary and they shall take precedence over the information contained in this document. Vaisala makes no representations on this document's compliance with the local rules and regulations applicable at any given time, and hereby disclaims any and all responsibilities related thereto. You are instructed to confirm the applicability of the local rules and regulations and their effect on the intended use of the products and services.

This document does not create any legally binding obligations for Vaisala towards customers or end users. All legally binding obligations are set forth exclusively in the applicable contract or in the relevant set of General Conditions of Vaisala (www.vaisala.com/policies).

This product contains software developed by Vaisala or third parties. Use of the software is governed by license terms and conditions included in the applicable contract or, in the absence of separate license terms and conditions, by the General License Conditions of Vaisala Group.

This product may contain open-source software (OSS) components. In the event this product contains OSS components, then such OSS is governed by the terms and conditions of the applicable OSS licenses, and you are bound by the terms and conditions of such licenses in connection with your use and distribution of the OSS in this product. Applicable OSS licenses are included in the product itself or provided to you on any other applicable media, depending on each individual product and the product items delivered to you.

Table of contents

1. About this document 5
1.1 Version information 5
1.2 Related manuals 5
1.3 Documentation conventions 5
1.4 Trademarks 6

2. Product overview 7
2.1 Vaisala Air Quality Transmitter AQT560 7
2.2 AQT560 measurements 8
2.3 Safety 8
2.3.1 Eye safety 10
2.3.2 ESD protection 11

3. Data reporting 12
3.1 Measurement output 12
3.2 Gas measurements 14
3.2.1 Conversion factors for gases 15
3.3 Particle measurements 15

4. Configuration 17
4.1 Interface overview 17
4.2 Connecting to AQT560 over RS-232 17
4.3 Configuring Modbus interface 20
4.4 Configuring ASCII CSV data sending 21
4.4.1 ASCII CSV data message 22
4.5 Changing temperature unit 24
4.6 Setting system time 25

5. Operation 26
5.1 Viewing gas measurement data 26
5.2 Viewing particle measurement data 26
5.3 Changing particle sampling interval 27
5.4 Checking air quality index information 27
5.5 Adjusting linear correction 28

6. CLI commands and parameters 32
6.1 CLI command reference 32
6.2 Setting date (date command) 33
6.3 Restoring default configuration (initconf command) 34
6.4 Printing LPC information (lpc command) 35
6.5 Printing measurements (meas command) 36
6.6 Clearing screen (motd command) 39
6.7 Restarting device (reboot command) 40
6.8 Changing configuration parameter (set command) 40
6.9 Printing value of configuration parameter (show command) 41
6.10 Printing device status (status command) 42
6.11 Printing Unix name (uname command) 43
6.12 Saving to non-volatile memory (write command) 44
6.13 CLI parameters 44

7. Modbus interface description 49
7.1 Supported Modbus protocols 49
7.2 Function codes 49
7.3 Device identification 49

7.4 Byte order....50
7.5 Modbus register addresses....51
7.6 Setting system time over Modbus interface....61

8. Troubleshooting....63
8.1 Checking device health information....63
8.2 Writing a problem report....64

Appendix A: M12 pinout and wiring....65

Appendix B: OSS licenses....66

Warranty....69

Technical support....69

Recycling....69

1. About this document

1.1 Version information

This document provides instructions for configuring Vaisala Air Quality Transmitter AQT560.

Table 1 Document versions (English)

Document code	Date	Description
M213026EN-A	March 2024	First version. For device software version 5.0: • Modbus version 1.0 • CLI version 1.0 • ASCII CSV output version 1.0

1.2 Related manuals



Some of these documents can be found in our product documentation portal at docs.vaisala.com.

Table 2 AQT560 manuals

Document code	Name
M213027EN	Vaisala Air Quality Transmitter AQT560 Product Description
M212982EN	Vaisala Air Quality Transmitter AQT560 Setup Guide
M213026EN	Vaisala Air Quality Transmitter AQT560 Configuration Guide
M212983EN	*Vaisala Air Quality Transmitter AQT560 Maintenance Guide*

1.3 Documentation conventions



WARNING! Warning alerts you to a serious hazard. If you do not read and follow instructions carefully at this point, there is a risk of injury or even death.

AKRIBIS · www.akribis.info · contacto@akribis.info



CAUTION! Caution warns you of a potential hazard. If you do not read and follow instructions carefully at this point, the product could be damaged or important data could be lost.



Highlights important information on using the product.



Gives information for using the product more efficiently.



Lists tools needed to perform the task.



Indicates that you need to take some notes during the task.

1.4 Trademarks

Vaisala® is a registered trademark of Vaisala Oyj.

Modbus® is a registered trademark of Schneider Automation Inc.

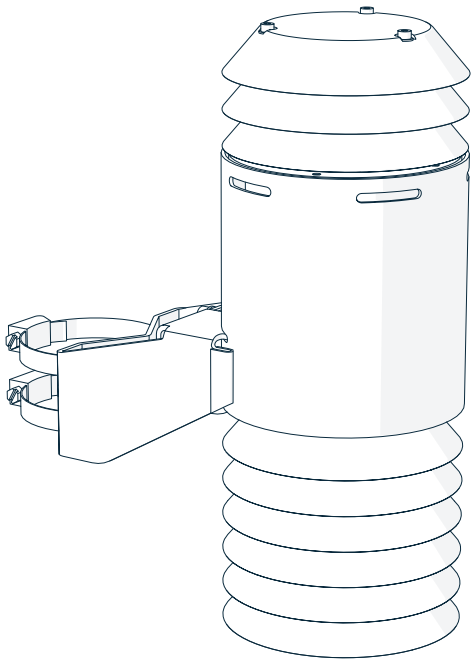
Windows® is either a registered trademark or trademark of Microsoft Corporation in the United States and other countries.

All other product or company names that may be mentioned in this publication are trade names, trademarks, or registered trademarks of their respective owners.

AKRIBIS · www.akribis.info · contacto@akribis.info

2. Product overview

2.1 Vaisala Air Quality Transmitter AQT560



AQT560 measures gaseous pollutants and particles in the ambient air. AQT560 is available in different models for measuring gases, particles, or both.

Thanks to its small weight, compact size, and good precision, AQT560 is ideal for supplementing existing air quality networks, for traffic management and road dust monitoring and mitigation, and for monitoring dust-emitting industrial facilities. Wildfire smoke monitoring is another area where AQT560 can be used effectively.

Table 3 AQT560 product models

Model	Measurements
Gases only	NO_2, NO, O_3, CO
Particles only	PM_{10}, $PM_{2.5}$, PM_1
Gas and particles	NO_2, PM_{10}, $PM_{2.5}$, PM_1
Gases and particles	NO_2, NO, O_3, CO, PM_{10}, $PM_{2.5}$, PM_1

2.2 AQT560 measurements

The measured variables have been configured to AQT560 at the factory.

Gas measurements are updated once a minute. The particle measurement update interval is user-configurable with a default of 10 minutes.

Table 4 AQT560 measurements

Measurement	Unit	Update interval	Reported value
Nitrogen dioxide ($NO_{2)}$	ppb, µg/m^3	1 minute	
Carbon monoxide (CO)	ppb, µg/m^3		
Ozone (O_3)	ppb, µg/m^3		
Nitrogen oxide (NO)	ppb, µg/m^3		
Particulate matter PM_{10}	µg/m^3	User-configurable Default: 10 minutes	1-minute average 1-hour average
Particulate matter $PM_{2.5}$	µg/m^3		
Particulate matter PM_1	µg/m^3		
Air temperature	°C/°F	1 minute	Instant
Air pressure	hPa		
Humidity	%RH		

The set of available measurements depends on the product model.

Stabilization period

The stabilization period is relevant for AQT560 devices that measure gases.

Let the unit stabilize for at least 24 hours before using the gas measurement results. During the stabilization period, the gas measurements are marked as invalid.

2.3 Safety

This product has been tested for safety. Note the following precautions:

WARNING! Do not replace components when the system is powered up. Disconnect all power sources before performing maintenance procedures.

WARNING! Do not substitute parts or modify the system, or install unsuitable parts in the system. Improper modification can damage the product or lead to malfunction.

WARNING! Assess the risks from the installation work. Consider also the effects of local weather conditions.

WARNING! Failure to comply with these precautions or with specific warnings elsewhere in these instructions violates safety standards of design, manufacture, and intended use of the product. Vaisala assumes no liability for the customer's failure to comply with these requirements.

WARNING! If the equipment is used in a manner not specified by Vaisala, the protection provided by the equipment may be impaired.

WARNING! Follow local and state legislation and regulations on occupational safety.

WARNING! Do not attempt to open the laser particle counter (LPC) for service. Removing the cover may cause exposure to harmful class 3B laser radiation and electrical shock.

CAUTION! Improper modification can damage the product or lead to malfunction. Any modification voids your warranty.

CAUTION! To prevent electrostatic discharge, avoid touching component contacts or connectors.



Wear personal protective equipment (PPE).

2.3.1 Eye safety

Vaisala Air Quality Transmitter AQT560 is classified as a Class 1 laser device in accordance with International Standard IEC 60825-1:2014 and EN 60825-1:2014+A11:2021. It complies with 21 CFR 1040.10 and 1040.11 except for conformance with IEC 60825-1 ED. 3., as described in the Laser Notice No. 56, dated May 8,2019. A Class 1 laser device is safe under all conditions of normal use.

AQT560 incorporates a Class 3B laser. The laser is contained in an enclosure, preventing direct physical access to laser radiation.

Table 5 Incorporated laser module

Property	Description/Value
Laser module	Class 3B
Laser wavelength	658 nm
Maximum power	22 mW

The device is equipped with the following laser product labels.

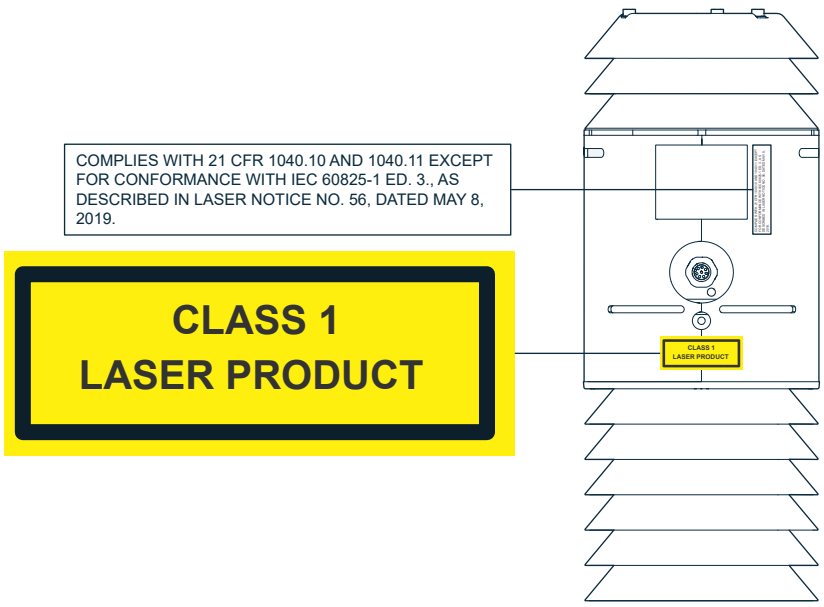


Figure 1 Location of Class 1 laser product labels on AQT560

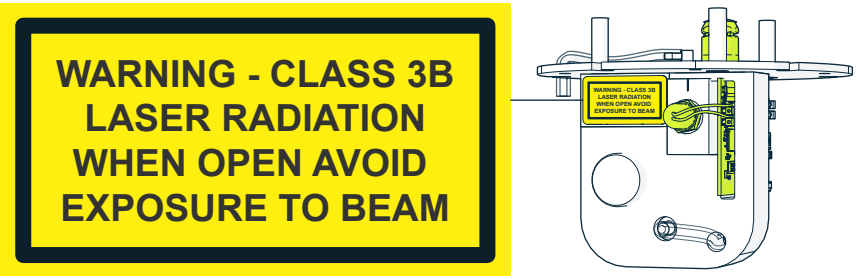


Figure 2 Location of Class 3B laser radiation label on AQT560 inside laser particle counter (LPC)

Follow the safety information to avoid exposure to laser radiation.

2.3.2 ESD protection

Electrostatic discharge (ESD) can damage electronic circuits. Vaisala products are adequately protected against ESD for their intended use. However, it is possible to damage the product by delivering electrostatic discharges when touching, removing, or inserting any objects in the equipment housing.

To avoid delivering high static voltages to the product:

- Handle ESD-sensitive components on a properly grounded and protected ESD workbench or by grounding yourself to the equipment chassis with a wrist strap and a resistive connection cord.
- If you are unable to take either precaution, touch a conductive part of the equipment chassis with your other hand before touching ESD-sensitive components.
- Hold component boards by the edges and avoid touching component contacts.

3. Data reporting

More information

- Measurement output (page 12)
- Gas measurements (page 14)
- Conversion factors for gases (page 15)
- Particle measurements (page 15)

3.1 Measurement output

The measurement data is available over RS-485 by means of Modbus (ASCII and RTU modes), or alternatively in ASCII CSV format.

The measurement data can be monitored also through CLI.

The following tables present the measurements, their units, and resolution.



The set of available gases depends on the electrochemical gas sensor setup.
Particle measurement results are included when a laser particle counter (LPC) is installed and enabled.

Table 6 Temperature

Value	Modbus	CLI	ASCII CSV
Air temperature	`0x000A`: 0.1 °C / 0.1 °F	• **`meas`**: °C / °F • **`meas`** `--metric`: °C / °F • **`meas`** `--ppb`: °C / °F	°C / °F

Table 7 Humidity

Value	Modbus	CLI	ASCII CSV
Air humidity	`0x000B`: 0.1 %RH	• **`meas`**: %RH • **`meas`** `--metric`: %RH • **`meas`** `--ppb`: %RH	%RH

Table 8 Pressure

Value	Modbus	CLI	ASCII CSV
Air pressure	`0x000C`: 0.1 hPa	• **`meas`**: hPa • **`meas`** `--metric`: hPa • **`meas`** `--ppb`: hPa	hPa

Table 9 Gas concentration

Value	Modbus	CLI	ASCII CSV
NO_2	• `0x0000`: ppb • `0x0066`: 0.1 µg/m³	• **meas**: ppm • **meas** `--metric`: µg/m³ • **meas** `--ppb`: ppb	ppm
SO_2	• `0x0001`: ppb • `0x0067`: 0.1 µg/m³	• **meas**: ppm • **meas** `--metric`: µg/m³ • **meas** `--ppb`: ppb	ppm
CO	• `0x0002`: ppb • `0x0068`: 1 µg/m³	• **meas**: ppm • **meas** `--metric`: µg/m³ • **meas** `--ppb`: ppb	ppm
H_2S	• `0x0004`: ppb • `0x0069`: 0.1 µg/m³	• **meas**: ppm • **meas** `--metric`: µg/m³ • **meas** `--ppb`: ppb	ppm
O_3	• `0x0005`: ppb • `0x006A`: 0.1 µg/m³	• **meas**: ppm • **meas** `--metric`: µg/m³ • **meas** `--ppb`: ppb	ppm
NO	• `0x0006`: ppb • `0x006B`: 0.1 µg/m³	• **meas**: ppm • **meas** `--metric`: µg/m³ • **meas** `--ppb`: ppb	ppm

Table 10 Mass concentration without linear correction (1-minute values)

Value	Modbus	CLI	ASCII CSV
PM_1	`0x0075`: 0.1 µg/m³	**meas** `--nolc`: µg/m³	–
$PM_{2.5}$	`0x0073`: 0.1 µg/m³	**meas** `--nolc`: µg/m³	–
PM_{10}	`0x0074`: 0.1 µg/m³	**meas** `--nolc`: µg/m³	–

Table 11 Mass concentration without linear correction (1-hour values)

Value	Modbus	CLI	ASCII CSV
PM_1	0x1011: 0.1 µg/m³	**meas** `--pm1hcsv`: µg/m³	–
$PM_{2.5}$	0x1012: 0.1 µg/m³	**meas** `--pm1hcsv`: µg/m³	–
PM_{10}	0x1013: 0.1 µg/m³	**meas** `--pm1hcsv`: µg/m³	–

Table 12 Mass concentration with linear correction (1-minute values)

Value	Modbus	CLI	ASCII CSV
PM_1	`0x0037`: 0.1 µg/m³	• **meas**: µg/m³ • **meas** `--metric`: µg/m³ • **meas** `--ppb`: µg/m³	µg/m³

Value	Modbus	CLI	ASCII CSV
$PM_{2.5}$	`0x0008`: 0.1 µg/m³	• **meas**: µg/m³ • **meas** `--metric`: µg/m³ • **meas** `--ppb`: µg/m³	µg/m³
PM_{10}	`0x0009`: 0.1 µg/m³	• **meas**: µg/m³ • **meas** `--metric`: µg/m³ • **meas** `--ppb`: µg/m³	µg/m³

Table 13 Mass concentration with linear correction (1-hour values)

Value	Modbus	CLI	ASCII CSV
PM_1	0x100E: 0.1 µg/m³	**meas** `--pm1hcsv`: µg/m³	–
$PM_{2.5}$	0x100F: 0.1 µg/m³	**meas** `--pm1hcsv`: µg/m³	–
PM_{10}	0x1010: 0.1 µg/m³	**meas** `--pm1hcsv`: µg/m³	–

3.2 Gas measurements

Gas measurements are available through Modbus, CLI, and ASCII CSV.

Table 14 Gas measurement data output

Value [1)]	Modbus	CLI
With linear correction in ppm	–	**meas**
With linear correction in µg/m³	• 0x0066–0x006B	**meas** `--metric`
With linear correction in ppb	• 0x0000–0x0002 • 0x0004–0x0006	**meas** `--ppb`
Without linear correction in ppb	• 0x006C–0x006E • 0x0070–0x0072	–
Without linear correction in ppm	–	**meas** `--nolc`
Stabilization flag	0x0033	**show** `meas_state`
Temperature flag	0x0034	–
Combined stabilization and temperature validity	0x001B	–

1) For each available gas measurement parameter.

For details, see the full list of Modbus register addresses.

For the µg/m^3 converted values and other options in CLI, see the description of the **`meas`** command.

More information

- ASCII CSV data message (page 22)
- Printing measurements (meas command) (page 36)
- Modbus register addresses (page 51)

3.2.1 Conversion factors for gases

Conversion from ppb to µg/m^3 is calculated with the following formula:

µg/m^3 = ppb * conversion_factor

Table 15 Conversion factors for conversion from ppb to µg/m^3

Gas	Conversion factor
NO_2	1.912
SO_2	2.66
CO	1.16
H_2S	1.417
O_3	2.00
NO	1.247

The conversion assumes a temperature of +20 °C (+68 °F) (European standard).

3.3 Particle measurements

Particle measurements are available through Modbus, CLI, and ASII CSV.

The values are 1-minute averages and 1-hour averages.

The unit is always µg/m^3.

Table 16 Particle measurement data output

Value		Modbus			CLI		
		PM_1	**$PM_{2.5}$**	**PM_{10}**	**PM_1**	**$PM_{2.5}$**	**PM_{10}**
1-minute average	With linear correction	0x0037	0x0008	0x0009	**meas** **meas** `--metric` **meas** `--ppb`		
	Without linear correction	0x0075	0x0073	0x0074	**meas** `--nolc`		
	Fog flag	0x007C	0x007D	0x007E	–		
1-hour average	With linear correction	0x100E	0x100F	0x1010	**meas** `--pm1hcsv`		
	Without linear correction	0x1011	0x1012	0x1013			
	Fog flag	0x1014	0x1015	0x1016			

For details, see the full list of Modbus register addresses.

For the values in CLI and the particle bins, use the CLI command **meas**. For details, see the description of the **meas** command.

The 1-hour average value is calculated hourly, starting from full hours, for example 00:00. When the calculation is ready after 60 minutes at 01:00, it is updated to all registers and the calculation of the next 1-hour average value begins. This means that the 1-hour average value reflects the past full hour and is updated every full hour.

The fog flag is raised for the 1-hour average value when at least 1 measurement value from that period is flagged. This means that the fog flag is active for at least 1 hour.

More information

- ASCII CSV data message (page 22)
- Printing measurements (meas command) (page 36)
- Modbus register addresses (page 51)

4. Configuration

4.1 Interface overview

AQT560 outputs data in the Modbus ASCII, Modbus RTU format (default), and ASCII CSV formats. You can use only 1 format at a time.

AQT560 has a command line interface (CLI) and a Modbus interface:

- Modbus interface properties are configured over the CLI interface.
- All measurement values are available from both CLI and Modbus.
- All configuration parameters are available through CLI, while Modbus supports the most important configuration parameters.

The CLI interface is for a local connection over RS-232.

Table 17 Data connection specifications

Property	Description/Value
Data output	Modbus® ASCII, Modbus® RTU, ASCII CSV
Serial data interface	RS-485
Maintenance interface [1)]	RS-232

1) Recommended Vaisala USB maintenance cable kit (253163SET).

More information

- Configuring Modbus interface (page 20)
- Configuring ASCII CSV data sending (page 21)
- Changing temperature unit (page 24)
- Setting system time (page 25)

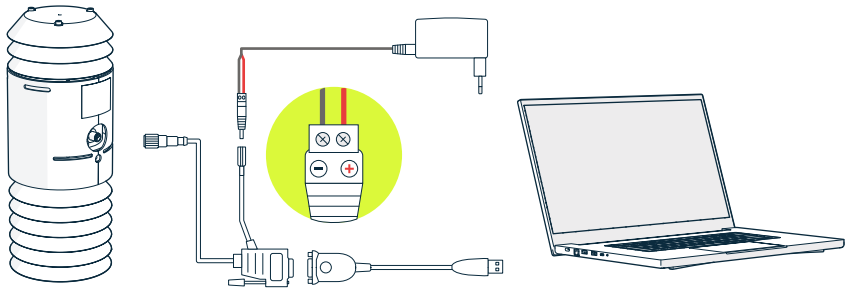
4.2 Connecting to AQT560 over RS-232

- Computer
- Service cable kit / Maintenance cable kit (253163SET), purchased separately from Vaisala
- Flat head screwdriver
- Terminal block
- Terminal program with a command line interface (CLI), such as PuTTY or TeraTerm.

You can establish a connection to AQT560 on the site to change settings, for example. To connect to AQT560, create a local maintenance connection over the serial RS-232 interface.

The instructions and examples here use and refer to PuTTY. You can also use another program with a command line interface, but the user interface looks different.

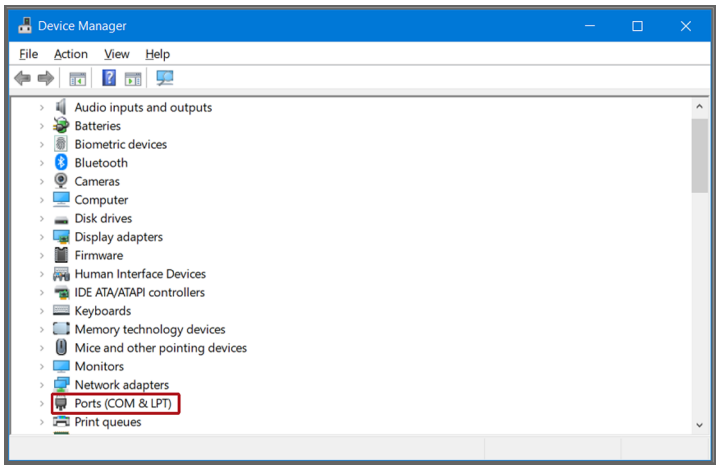
1. Connect your computer to AQT560 with the service cable kit.



Connect the open-ended cables of the power source to the adapter and attach the adapter to the DC power connector (10–25 V DC).
Alternatively, connect your own power source directly to the DC power connector.

Wire color	Power
Red	+
Black	–

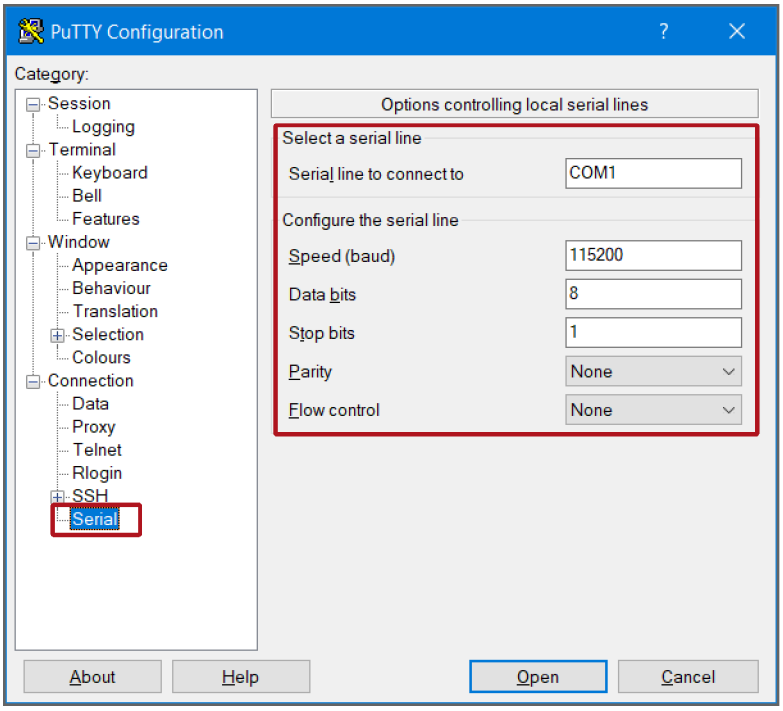
2. Open a terminal program (for example, PuTTY or TeraTerm).
3. On your computer, select **Start > Control Panel > Device Manager > Ports (COM & LPT)**.



In **Device Manager**, check to which port AQT560 is connected with the name **ATEN USB to Serial Bridge**. For example: ATEN USB to Serial Bridge (**COM1**).

4. In PuTTY, select **Serial** or **Serial & USB**.

> The screen may look different, depending on the computer and Windows operating system version.



5. In **PuTTY Configuration**, select the COM port where you connected ATEN USB to Serial Bridge.

   If **ATEN USB to Serial Bridge** is not visible as a COM port, download and install new **ATEN USB to Serial Bridge** drivers from the ATEN website.
   Use the following COM port settings.

   Table 18 COM port settings

Parameter	Value
Baud rate	115200
Data bits	8
Stop bits	1
Parity	None
Flow control	None

   These settings are applied on your computer so that it can connect to AQT560.

6. Select **Open** in the terminal program.

   Connection to AQT560 is now open and you can proceed to read and set parameters. To start, press **ENTER** or type command **`motd`**.

4.3 Configuring Modbus interface

The default communication settings are listed in the following table. Use the settings that are suitable for you.

Table 19 Modbus interface configuration parameters

Parameter	Type	Range	Default	Unit	Description
`rs485_mode`	uint8	• 0 = ASCII CSV • 1 = Modbus ASCII • 4 = Modbus RTU	4	–	RS-485 mode
`rs485_addr`	uint8	1–253	1	–	Modbus address (not used in ASCII CSV mode)
`rs485_baud`	uint32	• 4800 • 9600 • 19200 • 38400 • 57600 • 115200	19200	bps	RS-485 baud rate
`rs485_databits`	uint8	• 7 • 8	8	bits	RS-485 data bits
`rs485_parity`	Text	• `N` = None • `E` = Even • `O` = Odd	E	–	RS-485 parity Case insensitive
`rs485_stopbits`	uint8	• 1 • 2	1	bits	RS-485 stop bits

1. Establish a connection to the device.

2. Using a terminal program, define the settings for the RS-485 interface.
   - To use the RS-485 port in Modbus ASCII mode, type the following:

```
set rs485_baud=9600
set rs485_mode=1
set rs485_databits=7
set rs485_parity=e
set rs485_stopbits=1
write --really
```

     - Bit rate: 9600 bps
     - Mode: ASCII
     - Port settings: 7E1 (7 data bits, even parity, 1 stop bit)
   - To use the RS-485 port in Modbus RTU mode, type the following:

```
set rs485_baud=19200
set rs485_mode=4
set rs485_databits=8
set rs485_parity=e
set rs485_stopbits=1
write --really
```

     - Bit rate: 19200
     - Mode: RTU
     - Port settings: 8E1 (8 data bits, even parity, 1 stop bit)

The device is ready.

More information

- Changing configuration parameter (set command) (page 40)
- CLI parameters (page 44)

4.4 Configuring ASCII CSV data sending

The ASCII CSV output mode is an alternative to the Modbus API. It is a simplified measurement monitoring output for users that are only collecting data, without the ability to control and monitor the device otherwise.

The comma-separated values (CSV) output is sent automatically every 20 seconds. The output contains 1 measurement from all measurement outputs on 1 line and with a timestamp. The output data does not contain validity or device health information.

1. Establish a connection to the device.

2. Set the RS-485 interface to ASCII CSV mode using CLI.

```
set rs485_mode=0
set rs485_baud=115200
set rs485_databits=8
set rs485_parity=n
set rs485_stopbits=1
write --really
```

   - Port settings: 8N1 (8 data bits, no parity, 1 stop bit)

   Data sending starts automatically after selecting the mode.

3. Close the local RS-232 maintenance connection.

You can view the data through the RS-485 connection. For example, if the RS-485 port is connected to the serial port of your computer, the ASCII CSV data is available there.

More information

- Changing configuration parameter (set command) (page 40)
- CLI parameters (page 44)

4.4.1 ASCII CSV data message

The ASCII CSV data message is used for outputting measurement data.

```
<timestamp>,<conditions>,<gases>,<particles>,<config>,<uptime>
```

Table 20 ASCII CSV message data fields

Field	Description	Example
Timestamp	UTC date and time in ISO 8601 format	2020-12-11T03:00:41
Conditions	Environment conditions separated by commas (included always): • Air temperature (°C or °F, depending on the configuration [1]) • Air humidity (%RH) • Air pressure (hPa)	7.3,92.3,990.1
Gases	Gas measurement results in parts per million (ppm) separated by commas Possible gases, depending on electrochemical gas sensor setup: • NO_2, SO_2, CO, H_2S, O_3, NO Maximum number of electrochemical gas sensors at a time: 4	0.009,0.101,0.013,0.03

Field	Description	Example
Particles [2]	Particle measurement results in µg/m^3, separated by commas: • PM_1, $PM_{2.5}$, PM_{10} The particle measurement results are included when a laser particle counter is installed and enabled.	0.1,1.1,1.9
Config	Device setup separated by colons (:): • T = Air temperature (included always) • H = Air humidity (included always) • P = Air pressure (included always) • NO_2, SO_2, CO, H_2S, O_3, NO (included according to gas cell setup) • PM_1, $PM_{2.5}$, PM_{10} (included when LPC is installed and enabled) Order matches order of results in fields conditions, gases, and particles.	T:H:P:NO2:CO:O3:NO:PM1:PM2.5:PM10
Uptime	Time from last reboot in seconds.	2735641

1) *The temperature unit can be configured with CLI using* `tempunit` *parameter. For example, setting Celsius units with* `set tempunit=0`.
2) *When PM concentrations reach the upper measurement limit of the laser particle counter (LPC) due to some exceptional event, the PM readings may temporarily be negative and the LPC reboots itself. For example, the unloading of sand in a nearby construction site and the resulting sand dust may cause the particle concentration to be temporarily drastically different from normal. Because of this the device initially shows very high PM readings and then negative readings. The device typically recovers from these exceptional events back to normal operation.*

Example of CSV ASCII data message with 4 gas sensors and LPC

```
2022-01-22T07:37:38,22.3,24.1,999.3,0.182,2.920,0.575,0.140,0.1,1.1,1.9,T:H:P:NO2:CO
:O3:NO:PM1:PM2.5:PM10,3185
2022-01-22T07:38:38,22.3,24.1,999.3,0.170,2.921,0.551,0.131,0.1,1.1,1.9,T:H:P:NO2:CO
:O3:NO:PM1:PM2.5:PM10,3245
2022-01-22T07:39:38,22.3,24.1,999.3,0.159,2.919,0.527,0.123,0.1,1.1,1.9,T:H:P:NO2:CO
:O3:NO:PM1:PM2.5:PM10,3305
```

Example of CSV ASCII data message with 4 gas sensors, without LPC

```
2022-01-22T08:07:38,22.3,24.1,999.4,0.108,2.926,0.416,0.084,T:H:P:NO2:CO:O3:NO,4983
2022-01-22T08:08:38,22.3,24.0,999.4,0.101,2.927,0.402,0.079,T:H:P:NO2:CO:O3:NO,5043
2022-01-22T08:09:38,22.3,24.0,999.4,0.095,2.927,0.389,0.074,T:H:P:NO2:CO:O3:NO,5103
```

Example of CSV ASCII data message with LPC only

```
2022-01-22T07:40:38,22.4,24.1,999.3,0.1,1.1,1.9,T:H:P:PM1:PM2.5:PM10,3364
2022-01-22T07:41:38,22.4,24.1,999.3,0.1,1.1,1.9,T:H:P:PM1:PM2.5:PM10,3424
2022-01-22T07:42:38,22.4,24.1,999.3,0.1,1.1,1.9,T:H:P:PM1:PM2.5:PM10,3484
```

The actual message does not contain line feeds.

4.5 Changing temperature unit

Table 21 Temperature unit parameter

Parameter	Type	Range	Default	Unit	Description
`tempunit`	uint8	• `0` = °C • `1` = °F	0	–	Temperature unit for Modbus, ASCII CSV, and CLI interfaces

The Modbus register address for this operation is 0x001C.

1. Establish a connection to the device.
2. To change the temperature unit to Celsius with CLI, type:

```
set tempunit=0
write --really
```

To change to Fahrenheit, type:

```
set tempunit=1
write --really
```

3. To check that the settings are correct, type:

```
show tempunit
```

More information

- Changing configuration parameter (set command) (page 40)
- CLI parameters (page 44)

4.6 Setting system time

1. Establish a connection to the device.
2. To set the system date and time with CLI, type:

```
date [YYYY-MM-DDTHH:II:SS]
```

For example:

```
date 2020-05-22T12:34:11
```

3. To check the setting, type:

```
show time
```

More information

- Setting date (date command) (page 33)
- Setting system time over Modbus interface (page 61)

5. Operation

You can do the operations using a local connection with CLI commands and/or remotely over the Modbus interface.

To use the CLI commands, establish a local maintenance connection to the device over RS-232.

More information

- Viewing gas measurement data (page 26)
- Viewing particle measurement data (page 26)
- Changing particle sampling interval (page 27)
- Checking air quality index information (page 27)
- Adjusting linear correction (page 28)

5.1 Viewing gas measurement data

You can get the gas measurements with or without the linear correction with 1-hour average.

1. Depending on the output you are using, connect to the device to get the data.

More information

- Measurement output (page 12)
- Gas measurements (page 14)
- Modbus register addresses (page 51)

5.2 Viewing particle measurement data

You can get the particle measurements with or without the linear correction with 10-minute average or 1-hour average.

1. Depending on the output you are using, connect to the device to get the data.

More information

- Measurement output (page 12)
- Particle measurements (page 15)
- Modbus register addresses (page 51)

5.3 Changing particle sampling interval

Table 22 LPC measurement interval

Parameter	Type	Range	Default	Unit	Description
`lpc_int`	uint8	2–255	10	min	LPC measurement interval in minutes

The Modbus register address for this operation is 0x007F.

1. Establish a connection to the device.
2. To change the particle sampling interval with CLI, type:

```
set lpc_int <sampling interval>
```

A shorter sampling interval increases the power consumption.

3. To check that the settings are correct, type:

```
show lpc_int
```

More information

- Changing configuration parameter (set command) (page 40)
- CLI parameters (page 44)

5.4 Checking air quality index information

You can check the air quality index from the device.

The Modbus register addresses for this operation are:
- 0x0029
- 0x002A

1. Establish a connection to the device.
2. To check the air quality index with CLI, type:

```
$ show aqi
```

Example response:

```
show: US EPA Air Quality Index: 310, pollutant: PM25, 24h
```

The device reports the following information.

Table 23 Fields in response message

Field	Description	Range/Value
Reporting standard	Fixed value	`US EPA Air Quality`
`Index`	Air quality index	• 0–500 • –32768 = AQI calculation failed
`pollutant`	Pollutants affecting the air quality index	• `O3, 8h` = 8-h average for O_3 • `O3, 1h` = 1-h average for O_3 • `PM25, 24h` = 24-h average for $PM_{2.5}$ • `PM10, 24h` = 24-h average for PM_{10} • `CO, 8h` = 8-h average for CO • `SO2, 1h` = 1-h average for SO_2 • `SO2, 24h` = 24-h average for SO_2 • `NO2, 1h` = 1-h average for NO_2 • –32768 = AQI calculation failed

More information

- Printing value of configuration parameter (show command) (page 41)
- CLI parameters (page 44)

5.5 Adjusting linear correction

Span and zero values are used for the linear correction of the measured values. Zero parameter corrects the offset and span the sensitivity (gain).

You can set the zero and span correction for gas and particle measurements.

The adjusted values affect the measured values of the following Modbus register addresses.

To adjust the span and zero through the Modbus interface, use registers 0x0086–0x0097.

Table 24 Modbus registers for measurement data

Address	Description
0x0000–0x0006	Gas concentrations in parts per billion (ppb) with linear correction
0x0066–0x006B	Gas concentrations in µg/m^3 with linear correction
0x0008–0x0009 0x0037	Particle matter PM_1, $PM_{2.5}$, and PM_{10} in µg/m^3 1-minute average with linear correction
0x100E–0x1010	Particle matter PM_1, $PM_{2.5}$, and PM_{10} in µg/m^3 1-hour average with linear correction

After making these changes, the device outputs data that is corrected by these zero and span correction factors.

Table 25 Linear correction parameters for gases

Parameter	Type	Range	Default	Unit	Description
`co_zero`	int16	–10000 … 10000	0	ppb	CO sensor zero correction
`co_span`	uint8	1–255	100	%	CO sensor span correction (1–255 = 0.01–2.55)
`no_zero`	int16	–10000 … 10000	0	ppb	NO sensor zero correction
`no_span`	uint8	1–255	100	%	NO sensor span correction (1–255 = 0.01–2.55)
`no2_zero`	int16	–10000 … 10000	0	ppb	NO_2 sensor zero correction
`no2_span`	uint8	1–255	100	%	NO_2 sensor span correction (1–255 = 0.01–2.55)
`so2_zero`	int16	–10000 … 10000	0	ppb	SO_2 sensor zero correction
`so2_span`	uint8	1–255	100	%	SO_2 sensor span correction (1–255 = 0.01–2.55)
`h2s_zero`	int16	–10000 … 10000	0	ppb	H_2S sensor zero correction
`h2s_span`	uint8	1–255	100	%	H_2S sensor span correction (1–255 = 0.01–2.55)
`o3_zero`	int16	–10000 … 10000	0	ppb	O_3 sensor zero correction

Parameter	Type	Range	Default	Unit	Description
o3_span	uint8	1–255	100	%	O_3 sensor span correction (1–255 = 0.01–2.55)

Table 26 Linear correction parameters for particles

Parameter	Type	Range	Default	Unit	Description
pm1_zero	int16	−10000 … 10000	0	0.1 µg/m³	PM_1 offset (zero) correction (−10000 … 10000 = −1000.0 … 1000.0)
pm1_span	int16	1 … 10000	1000	‰	PM_1 span (gain) correction (1 … 10000 = 0.001 … 10.000)
pm25_zero	int16	−10000 … 10000	0	0.1 µg/m³	$PM_{2.5}$ offset (zero) correction (−10000 … 10000 = −1000.0 … 1000.0)
pm25_span	int16	1 … 10000	1000	‰	$PM_{2.5}$ span (gain) correction (1 … 10000 = 0.001 … 10.000)
pm10_zero	int16	−10000 … 10000	0	0.1 µg/m³	PM_{10} offset (zero) correction (−10000 … 10000 = −1000.0 … 1000.0)
pm10_span	int16	1 … 10000	1000	‰	PM_{10} span (gain) correction (1 … 10000 = 0.001 … 10.000)

1. Establish a connection to the device.
2. Adjust the gain and offset of the parameters.

   For example, to adjust offset up by 100 ppb and sensitivity 1.2 times (120 %) higher, type:

```
set o3_zero=100
set o3_span=120
```

3. To save the changes:

```
write --really
```

   The new values overwrite the previous zero and span values.

Values without linear correction are available with Modbus registers 0x006C–0x0075.

More information

- CLI parameters (page 44)
- Changing configuration parameter (set command) (page 40)

6. CLI commands and parameters

More information

- CLI command reference (page 32)
- CLI parameters (page 44)

6.1 CLI command reference

The commands are for use in a standard terminal program (command-line interface, CLI).

The CLI commands include the following:

- Device identification commands: **motd**, **uname**
- System control command: **reboot**
- Configuration commands: **date**, **initconf**, **set**, **show**, **write**
- Measurement output command: **meas**
- LPC identification command: **lpc**
- Device status command: **status**

Table 27 List of commands

Command	Description	Reference
date [<YYYY-MM-DDThh:mm:ss>]	Prints or changes current date and time.	Setting date (**date** command) (page 33)
initconf --really	Loads default configuration, including: • Communication settings • Configuration and linear correction	Restoring default configuration (**initconf** command) (page 34)
lpc --info	Shows LPC module information, including: • Serial number • Firmware version	Printing LPC information (**lpc** command) (page 35)
meas [--metric|--ppb|--nolpc]	Outputs current measurements: • Gas measurements with linear correction in metric (µg/m^3) units • Gas measurements with linear correction in parts per billion (ppb) units • Gas measurements without linear correction in parts per million (ppm) units	Printing measurements (**meas** command) (page 36)
motd	Clears screen and prints a welcome banner.	Clearing screen (**motd** command) (page 39)

Command	Description	Reference
reboot --really	Restarts the device. If gas cells are in use, rebooting the device restarts the 24 hour stabilization time.	Restarting device (**reboot** command) (page 40)
set <parameter>= <value>	Changes a configuration parameter.	Changing configuration parameter (**set** command) (page 40)
show <parameter>	Prints the value of a configuration parameter.	Printing value of configuration parameter (**show** command) (page 41)
status	Prints the device status and possible reason.	Printing device status (**status** command) (page 42)
uname [-a]	Prints the Unix name of the system (short or long name).	Printing Unix name (**uname** command) (page 43)
write --really	Writes the configuration changes to non-volatile memory of the device.	Saving to non-volatile memory (**write** command) (page 44)

To use the commands, establish a connection to the device.

6.2 Setting date (**date** command)

To display or change the device date, use the **date** command.

```
date [YYYY-MM-DDThh:mm:ss]
```

Table 28 Parameters for **date** command

Parameter	Value	Mandatory	Description
(none)	–	–	Shows the current date and time in ISO 8601 format.
YYYY-MM-DDThh:mm:ss	Year, month, day, hour, minute, second	No	Sets a new date. T is a date/time separator.

Example of changing the date:

```
date 2020-12-02T13:50:36
```

Example response:

```
Date updated to 2020-12-02T13:50:36
$
```

More information

- Setting system time (page 25)

6.3 Restoring default configuration (**initconf** command)

To restore the default configuration, use the **initconf** command.

```
initconf --really
```

Table 29 Parameters for **initconf** command

Parameter	Value	Mandatory	Description
--really	Fixed	Yes	Loads the default configuration, including: • Communication settings • Configuration and linear correction

The command does not affect the following configurations:
- Device identification information
- Factory calibration information

The command does not affect connected devices, such as laser particle counter (LPC) and the temperature and humidity (TH) probe.

Example response:

```
Loading default configuration... OK!
Write the default configuration to non-volatile memory by using "write"
command.
$
```

Use the **write** command to save the configuration.

```
write --really
```

6.4 Printing LPC information (**lpc** command)

To display information about the laser particle counter (LPC), use the **lpc** command.

```
lpc --info
```

```
lpc --model
```

Table 30 Parameters for **lpc** command

Parameter	Value	Mandatory	Description
`--info`	Fixed	Yes	Shows LPC module information: • Serial number • Firmware version
`--model`	Fixed	Yes	Shows the model name of the LPC module.



The LPC must be connected to the system.

Querying information about the LPC module:

```
lpc --info
```

Example response 1:

```
Serial: B3245009
SW ver: 3.0.243.ad102e1
$
```

Example response 2 (no LPC connected):

```
No LPC option installed!
$
```

Querying information about the model of the LPC module.

```
lpc --model
```

Example response:

```
$ lpc --model
LPC model: LPC300
$
```

6.5 Printing measurements (**meas** command)

To display the measurements with specific units, use the **meas** command.

```
meas [--metric|--ppb|--nolc|--pm1hcsv|--pncsv]
```

Table 31 Parameters for **meas** command

Parameter	Value	Mandatory	Description
(none)	–	No	Prints out all measurements with: • Gases in parts per million units (ppm) • With linear correction
`--metric`	Fixed	No	Prints out all measurements with: • Gases and particles in metric units (µg/m^3) • With linear correction
`--ppb`	Fixed	No	Prints out all measurements with: • Gases in parts per billion units (ppb) • Particles in metric units (µg/m^3) • With linear correction
`--nolc`	Fixed	No	Prints out all measurements: • With gases in parts per million units (ppm) • With particles in metric units (µg/m^3) • Without linear correction

Parameter	Value	Mandatory	Description

`--pm1hcsv`	Fixed	No	Prints out CSV report of 1-hour averages and corresponding fog flags for PM_1, $PM_{2.5}$, and PM_{10}.
			The output format is:
			PM1_1H,PM2P5_1H,PM10_1H,PM1_1H_RAW,PM2P5_1H_RAW,PM10_1H_RAW,PM1_FLG,PM2P5_FLG,PM10_FLG
			The output contains the following values:
			• Linear corrected 1-hour averages in µg/m^3
			• Raw 1-hour averages in µg/m^3
			• Fog flags associated with 1-hour average values:
			  • `0`: No fog detected during the 1-hour period.
			  • `1`: Fog detected during the 1-hour period.
			The 1-hour timing is synchronized with the calendar clock.
			The average value is published on every hour and stays constant during the whole hour. The new average is calculated in the background.
			The fog flag is set if at least 1 particle measurement sample during the 1-hour sampling period is flagged.

`--pncsv`	Fixed	No	Prints out measured particles in particle number bins 1–7 in particles per liter in CSV format. The particle number bins 1–7 contain numbers for the following particle sizes (in that order): • 0.6 µm • 1.0 µm • 1.6 µm • 2.5 µm • 4.0 µm • 6.3 µm • 10 µm

```
meas
```

Example response:

```
NO2 (ppm): 0.004
SO2 (ppm): 0.562
CO (ppm): 0.077
O3 (ppm): -0.002
PM1 (ug/m3): 0.1
PM2.5 (ug/m3): 0.4
PM10 (ug/m3): 2.2
TEMP (C): 22.2
HUM (%RH): 31.2
PRES (mbar): 1012.4
Uptime (s): 10850
$
```

Example of setting measurement output to use metric (µg/m^3) units.

```
meas --metric
```

Example response:

```
NO2 (ug/m3): 5.9
SO2 (ug/m3): 912.5
CO (ug/m3): 63.6
O3 (ug/m3): -1.8
PM1 (ug/m3): 0.1
PM2.5 (ug/m3): 0.4
PM10 (ug/m3): 2.2
TEMP (C): 22.1
HUM (%RH): 31.5
PRES (mbar): 1012.3
Uptime (s): 10684
$
```

Example of setting measurement output to use parts per billion (ppb) units.

```
meas --ppb
```

Example response:

```
NO2 (ppb): 4.5
SO2 (ppb): 698.6
CO (ppb): 97.5
O3 (ppb): -0.4
PM1 (ug/m3): 0.1
PM2.5 (ug/m3): 0.4
PM10 (ug/m3): 2.2
TEMP (C): 22.3
HUM (%RH): 30.9
PRES (mbar): 1012.4
Uptime (s): 10923
$
```

Example of setting measurement output to print without linear correction, using parts per million (ppm) units.

```
meas --nolc
```

Example response:

```
NO2 (ppm): 0.004
SO2 (ppm): 0.562
CO (ppm): 0.077
O3 (ppm): -0.002
PM1 (ug/m3): 0.2
PM2.5 (ug/m3): 0.4
PM10 (ug/m3): 2.2
TEMP (C): 22.2
HUM (%RH): 31.2
PRES (mbar): 1012.4
Uptime (s): 10850
$
```

Example of CSV report with 1-hour PM averages and fog flags.

```
meas --pm1hcsv
```

Example response:

```
0.5,2.3,0.2,0.4,2.2,0.1,0,0,0
$
```

Example of measured particles in particle number bins.

```
meas --pncsv
```

Example response:

```
0,2421,3127,3355,3404,3422,3426
$
```

More information

- Gas measurements (page 14)
- Particle measurements (page 15)

6.6 Clearing screen (**motd** command)

To clear the screen and print a welcome text, use the **motd** command.

```
motd
```

Example response:

```
Vaisala Air Quality Transmitter AQT560
FW version 5.0.0.d5535d2
Copyright (c) 2024 Vaisala Oyj

$
```

6.7 Restarting device (**reboot** command)

To restart the device, use the **reboot** command.

```
reboot --really
```

Table 32 Parameters for **reboot** command

Parameter	Value	Mandatory	Description
--really	Fixed	Yes	Confirms the restart.

Example response

```
$ reboot --really
Resetting device in 5 seconds...



Vaisala Air Quality Transmitter AQT560
FW version 5.0.0.d5535d2
Copyright (c) 2024 Vaisala Oyj



$
```

6.8 Changing configuration parameter (**set** command)

To change device configuration, use the **set** command.

```
set <parameter>=<value>
```

Table 33 Parameters for **set** command

Parameter	Mandatory	Description
For the list of parameters, see CLI parameters (page 44).	Yes	Changes the value of selected parameter.

Example of changing the baud rate:

```
set rs485_baud=19200
```

Example response:

```
set: rs485_baud=19200
$
```

For more information, see the full list of CLI parameters.

More information

- Configuring Modbus interface (page 20)
- Configuring ASCII CSV data sending (page 21)
- Changing temperature unit (page 24)
- Changing particle sampling interval (page 27)
- Adjusting linear correction (page 28)

6.9 Printing value of configuration parameter (**show** command)

To view the value of a configuration parameter, use the **show** command.

```
show <parameter>
```

Table 34 Parameters for **show** command

Parameter	Mandatory	Description
For the list of parameters, see CLI parameters (page 44).	Yes	Shows the value of selected parameter.

Example of checking device information:

```
show unit
```

Example response:

```
$ show unit
show: Unit configuration

Model: Vaisala AQT560
Serial: B0220002
Configuration: LPC only
$
```

For more information, see the full list of CLI parameters.

More information

- CLI parameters (page 44)

6.10 Printing device status (**status** command)

To print the device status, use the **status** command.

```
status
```

The following responses are possible:

- `Unknown (initializing)`
- `OK`
- `Degraded:` [reason]
- `Faulty:` [reason]

Example responses:

```
Device status:
OK
```

```
Device status:
Faulty: LPC malfunction
```

6.11 Printing Unix name (**uname** command)

To display the Unix name of the system, use the **uname** command.

```
uname [-a]
```

Table 35 Parameters for **uname** command

Parameter	Value	Mandatory	Description
(none)	–	–	Prints the Unix name of the system in short format.
-a	Fixed	No	Prints the Unix name of the system in long format.

Querying short name:

```
uname
```

Example response:

```
$ uname
Vaisala aqt-fw

$
```

Querying long name:

```
uname -a
```

Example response:

```
$ uname -a
Vaisala aqt-fw 5.0.0.74da4b4 2024-02-07T10:24:32Z AQT560BASECB-B avr-m2560/328p

$
```

6.12 Saving to non-volatile memory (**write** command)

To write the configuration changes to the non-volatile memory of the device, use the **write** command.

```
write --really
```

Use the **write** command together with the other commands, for example **set**.

Table 36 Parameters for **write** command

Parameter	Value	Mandatory	Description
`--really`	Fixed	Yes	Saves the configuration.

Example response:

```
Writing current configuration to non-volatile memory... 1248 bytes written!
$
```

6.13 CLI parameters

The following parameters are available for the air quality transmitter. They include a set of read-only and read/write parameters.

Read/write parameters:

- Modbus interface configuration parameters
- Temperature unit parameter
- LPC measurement interval
- Linear correction parameters for gases
- Linear correction parameters for particles

Read-only parameters:

- Device identification parameters
- Base module identification parameters
- Base module monitoring parameters
- Gas measurement configuration and diagnostics parameters
- LPC parameters
- Temperature and humidity probe (TH) probe parameters

Use the parameters with commands **`set`** (read/write parameters) and **`show`** (read-only and read/write parameters).

Table 37 Modbus interface configuration parameters

Parameter	Type	Range	Default	Unit	Description
`rs485_mode`	uint8	• 0 = ASCII CSV • 1 = Modbus ASCII • 4 = Modbus RTU	4	–	RS-485 mode
`rs485_addr`	uint8	1–253	1	–	Modbus address (not used in ASCII CSV mode)
`rs485_baud`	uint32	• 4800 • 9600 • 19200 • 38400 • 57600 • 115200	19200	bps	RS-485 baud rate
`rs485_databits`	uint8	• 7 • 8	8	bits	RS-485 data bits
`rs485_parity`	Text	• `N` = None • `E` = Even • `O` = Odd	E	–	RS-485 parity Case insensitive
`rs485_stopbits`	uint8	• 1 • 2	1	bits	RS-485 stop bits

Table 38 Temperature unit parameter

Parameter	Type	Range	Default	Unit	Description
`tempunit`	uint8	• `0` = °C • `1` = °F	0	–	Temperature unit for Modbus, ASCII CSV, and CLI interfaces

Table 39 LPC measurement interval

Parameter	Type	Range	Default	Unit	Description
`lpc_int`	uint8	2–255	10	min	LPC measurement interval in minutes

Table 40 Linear correction parameters for gases

Parameter	Type	Range	Default	Unit	Description
`co_zero`	int16	–10000 … 10000	0	ppb	CO sensor zero correction
`co_span`	uint8	1–255	100	%	CO sensor span correction (1–255 = 0.01–2.55)

Parameter	Type	Range	Default	Unit	Description
no_zero	int16	–10000 … 10000	0	ppb	NO sensor zero correction
no_span	uint8	1–255	100	%	NO sensor span correction (1–255 = 0.01–2.55)
no2_zero	int16	–10000 … 10000	0	ppb	NO_2 sensor zero correction
no2_span	uint8	1–255	100	%	NO_2 sensor span correction (1–255 = 0.01–2.55)
so2_zero	int16	–10000 … 10000	0	ppb	SO_2 sensor zero correction
so2_span	uint8	1–255	100	%	SO_2 sensor span correction (1–255 = 0.01–2.55)
h2s_zero	int16	–10000 … 10000	0	ppb	H_2S sensor zero correction
h2s_span	uint8	1–255	100	%	H_2S sensor span correction (1–255 = 0.01–2.55)
o3_zero	int16	–10000 … 10000	0	ppb	O_3 sensor zero correction
o3_span	uint8	1–255	100	%	O_3 sensor span correction (1–255 = 0.01–2.55)

Table 41 Linear correction parameters for particles

Parameter	Type	Range	Default	Unit	Description
pm1_zero	int16	–10000 … 10000	0	0.1 µg/m³	PM_1 offset (zero) correction (–10000 … 10000 = –1000.0 … 1000.0)
pm1_span	int16	1 … 10000	1000	‰	PM_1 span (gain) correction (1 … 10000 = 0.001 … 10.000)
pm25_zero	int16	–10000 … 10000	0	0.1 µg/m³	$PM_{2.5}$ offset (zero) correction (–10000 … 10000 = –1000.0 … 1000.0)
pm25_span	int16	1 … 10000	1000	‰	$PM_{2.5}$ span (gain) correction (1 … 10000 = 0.001 … 10.000)

Parameter	Type	Range	Default	Unit	Description
`pm10_zero`	int16	–10000 … 10000	0	0.1 µg/m³	PM_{10} offset (zero) correction (–10000 … 10000 = –1000.0 … 1000.0)
`pm10_span`	int16	1 … 10000	1000	‰	PM_{10} span (gain) correction (1 … 10000 = 0.001 … 10.000)

Table 42 Device identification parameters (read-only)

Parameter	Type	Range	Unit	Description
`serial`	Text	–	–	Device serial number

Table 43 Base module identification parameters (read-only)

Parameter	Type	Range	Unit	Description
`base_serial`	Text	–	–	Serial number of base control board with electrochemical gas sensors
`boardid`	Text	–	–	Serial number of base control board without electrochemical gas sensors
`cal_date`	Text	–	–	Base module calibration date in ISO 8601 format (YYYY-MM-DD) Combination of parameter values: • `cal_year` • `cal_month` • `cal_day`
`cal_day`	uint16	1–31	DD	Base module calibration day of month
`cal_month`	uint16	1–12	MM	Base module calibration month
`cal_year`	uint16	2000–2100	YYYY	Base module calibration year
`hwver`	Text	–	–	Base module hardware version
`swver`	Text	–	–	Base module firmware version

Table 44 Base module monitoring parameters (read-only)

Parameter	Type	Range	Unit	Description
`aqi`	Text			Air quality index (AQI) report
`model`	Text	–	–	Device model
`time`	Text	–	–	Current timestamp in ISO 8601 format

Parameter	Type	Range	Unit	Description
`unit`	Text	–	–	Combination of • Device model • Serial number • Configuration

Table 45 Gas measurement diagnostics parameters (read-only)

Parameter	Type	Range	Unit	Description
`health`	uint8	0–100	%	Device health status of electrochemical gas sensors in the device
`meas_state`	uint8	• `0` = Stabilization of electrochemical gas sensors ongoing • `1` = Stabilization of electrochemical gas sensors finished	–	Stabilization status of electrochemical gas sensors During stabilization of electrochemical gas sensors, measurement values are invalid

Table 46 LPC parameters (read-only)

Parameter	Type	Range	Unit	Description
`lpc_serial`	Text	–	–	Serial number of laser particle counter (LPC) module

Table 47 TH probe parameters (read-only)

Parameter	Type	Range	Unit	Description
`hmp_serial`	Text	–	–	HMP110 serial number
`hmp_version`	Text	–	–	HMP110 firmware version number

More information

- Configuring Modbus interface (page 20)
- Configuring ASCII CSV data sending (page 21)
- Changing temperature unit (page 24)
- Changing particle sampling interval (page 27)
- Adjusting linear correction (page 28)
- Printing value of configuration parameter (show command) (page 41)

7. Modbus interface description

7.1 Supported Modbus protocols

AQT560 interface supports:

- Modbus ASCII mode
- Modbus RTU mode

For the general full Modbus specification, see http://www.modbus.org/.

7.2 Function codes

AQT560 supports the following Modbus function codes.

Function code	Function name
0x03	Read multiple registers
0x06	Write single register
0x04	Read device ID

7.3 Device identification

AQT560 supports the following device identification objects.

QT560 supports only individual access of device identification objects (read device ID code 0x04).

Object ID	Category	Description	Content
0x00	Basic	Vendor name	Vaisala
0x01	Basic	Product code	AQT560
0x02	Basic	Version	For example: 5.0.0.d5535d2
0x03	Regular	Vendor URL	https://www.vaisala.com
0x04	Regular	Product name	Vaisala Air Quality Transmitter AQT560
0x05	Regular	Model name	For example: Model: CO, NO_2, NO, O_3, LPC

Object ID	Category	Description	Content
0x80	Extended	Serial number	For example: A0110001
0x81	Extended	Calibration date	For example: 2024-01-02
0x85	Extended	Hardware version	For example: B

7.4 Byte order

Individual registers follow the Modbus standard big-endian byte order for 16-bit values.

Values greater than 16 bits comprise multiple registers. These values are organized as follows.

32-bit values

In this application programming interface (API), 32-bit values are supported by the means of 2 consecutive 16-bit registers. The byte order for the 32-bit values is middle-endian: bytes within words are in big-endian order, while words in multi-word values are in little-endian order.

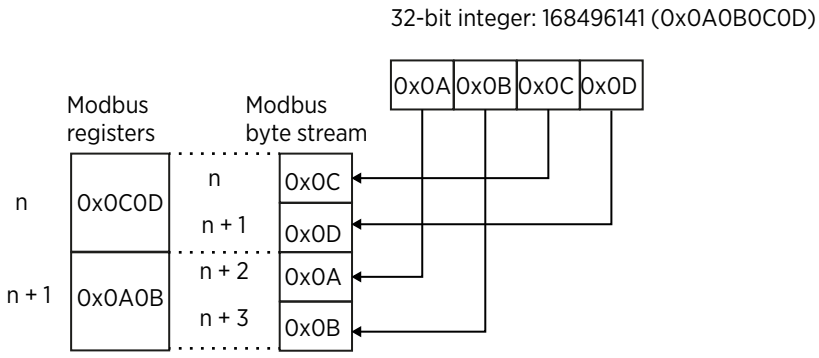


Figure 3 Byte order for 32-bit values

String fields

Several string type fields are available in this API, for example serial numbers. String fields consist of 1-byte ASCII characters packed into multiple 16-bit registers, 2 characters each.

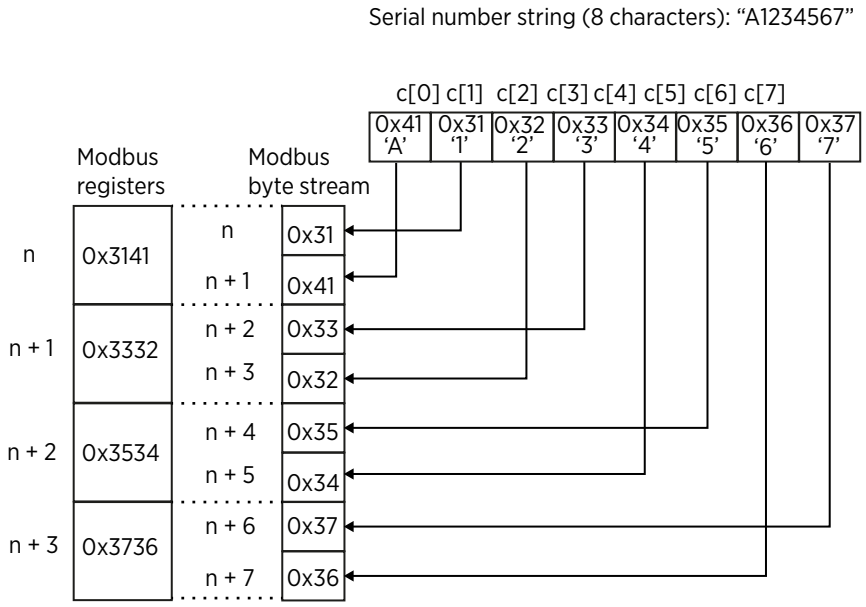


Figure 4 Byte order for string fields

7.5 Modbus register addresses

The following register addresses are available in AQT560. The register addresses are:

- R = Read only
- RW = Read/write
- W = Write only

When you change the device configuration settings through the Modbus interface, use Modbus register 0x00FE to store the new settings to the non-volatile memory, otherwise the new configuration is lost after reboot.

For example, to change the particle measurement interval to 2 minutes through Modbus:

1. Write value **2** to register 0x007F.
2. Write value **1** to register 0x00FE (save changes to non-volatile memory).

Table 48 Modbus register addresses 0x0000–0x00FE

Address	RW	Type	Unit	Register count	Description
0x0000	R	int16	ppb	1	NO_2 concentration in parts per billion (ppb) with linear correction • For a value without linear correction, use register 0x006C • For µg/m^3 value, use register 0x0066

Address	RW	Type	Unit	Register count	Description
0x0001	R	int16	ppb	1	SO_2 concentration in parts per billion (ppb) with linear correction • For a value without linear correction, use register 0x006D • For µg/m^3 value, use register 0x0067
0x0002	R	int16	ppb	1	CO concentration in parts per billion (ppb) with linear correction • For a value without linear correction, use register 0x006E • For µg/m^3 value, use register 0x0068
0x0003	-	-	-	-	Not in use
0x0004	R	int16	ppb	1	H_2S concentration in parts per billion (ppb) with linear correction • For a value without linear correction, use register 0x0070 • For µg/m^3 value, use register 0x0069
0x0005	R	int16	ppb	1	O_3 concentration in parts per billion (ppb) with linear correction • For a value without linear correction, use register 0x0071 • For µg/m^3 value, use register 0x006A
0x0006	R	int16	ppb	1	NO concentration in parts per billion (ppb) with linear correction • For a value without linear correction, use register 0x0072 • For µg/m^3 value, use register 0x006B
0x0007	-	-	-	-	Not in use
0x0008	R	int16	0.1 µg/m^3	1	Particulate matter $PM_{2.5}$ with linear correction [1)] If the $PM_{2.5}$ value is invalid, the register returns -32768. • For a value without linear correction, use register 0x0073
0x0009	R	int16	0.1 µg/m^3	1	Particulate matter PM_{10} [1)] If the PM_{10} value is invalid, the register returns −32768. • For a value without linear correction, use register 0x0074
0x000A	R	int16	0.1 °C / 0.1 °F	1	Air temperature (HMP110) in Celsius or Fahrenheit, depending on configuration Current configuration available from register 0x001C
0x000B	R	int16	0.1 %RH	1	Air humidity (HMP110)

Address	RW	Type	Unit	Register count	Description
0x000C	R	int16	0.1 hPa	1	Air pressure from an on-board analog pressure sensor
0x000D–0x0012	-	-	-	-	Not in use
0x0013	R	uint16	y	1	Calibration year
0x0014	R	uint16	mo	1	Calibration month
0x0015	R	uint16	d	1	Calibration day
0x0016	R	uint16	-	1	Unit configuration (bits/description)
					Bits / **Description**
					15:2 / Reserved
					1 / LPC configuration: • 0 = No LPC • 1 = LPC
					0 / Reserved
0x0017–0x0019	-	-	-	-	Not in use
0x001A	R	int16	s	1	Seconds elapsed after last measurement data update

0x001B	R	int16	-	1	Gas measurement validity
					• 0 = Measurement is not valid
					• 1 = Measurement is valid, including:
					  • 24-hour stabilization time passed (0x0033)
					  • Temperature is within valid range: < 38.0 °C / 100 °F (0x0034)

0x001C	R	uint16	-	1	Temperature unit configuration • 0 = Celsius (default) • 1 = Fahrenheit
0x001D–0x001E	-	-	-	-	Not in use
0x001F	R	int16	%	1	Device health index, which is the combined percentage of usage of electrochemical gas sensors; the value decreases from 100 % • 100 % = All electrochemical gas sensors are fully healthy • 0 % = At least one electrochemical gas sensor is over-aged
0x0020–0x0028	-	-	-	-	Not in use
0x0029	R	int16	AQI	1	• 0–500 = Air quality index • −32768 = AQI calculation failed AQI is calculated according to the documentation in Wikipedia: https://en.wikipedia.org/wiki/Air_quality_index#Computing_the_AQI

Address	RW	Type	Unit	Register count	Description

0x002A	R	int16	-	1	• 0–7 = Air quality index criteria
					  • 0 = O_3 8-h average
					  • 1 = O_3 1-h average
					  • 2 = $PM_{2.5}$ 24-h average
					  • 3 = PM_{10} 24-h average
					  • 4 = CO 8-h average
					  • 5 = SO_2 1-h average
					  • 6 = SO_2 24-h average
					  • 7 = NO_2 1-h average
					• –32768 = AQI calculation failed

0x002B–0x0032	-	-	-	-	Not in use
0x0033	R	int16	-	1	Stabilization invalidation flag for electrochemical gas sensors • 0 = 24-h stabilization time after power-up has passed • 1 = 24-h stabilization time after power-up not passed and measurement is invalid
0x0034	R	int16	-	1	Temperature invalidation flag for electrochemical gas sensors • 0 = Electrochemical gas sensor temperature valid • 1 = Electrochemical gas sensor temperature too high (≥ 38.0 °C / 100 °F) and measurement invalid
0x0035	-	-	-	-	Not in use
0x0036	R	int16	-	1	Humidity compensation (gas compensation) • 0 = Humidity compensation disabled • 1 = Humidity compensation enabled
0x0037	R	int16	0.1 µg/m^3	1	Particulate matter PM_1 with linear correction [1)] • For a value without linear correction, use register 0x0075 If the PM_1 value is invalid, the register returns –32768.
0x0038–0x004A	-	-	-	-	Not in use
0x004B	R	int16	-	1	Status reported by device • 0 = Unknown. Status information not yet available, starting up • 1 = Ok. Operating normally • 2 = Degraded. Operating but functionality degraded • 3 = Faulty For more information, see status code from register 0x004C

Address	RW	Type	Unit	Register count	Description	
0x004C	R	int16	-	1	Status codes • 0 = No specific status to report • 1 = LPC malfunction • 2 = HMP110 malfunction	
0x004D–0x004E	-	-	-	-	Not in use	
0x004F	R	uint16	-	1	AQT base module firmware version major number	
0x0050	R	uint16	-	1	AQT base module firmware version minor number	
0x0051	R	uint16	-	1	AQT base module firmware version build number	
0x0052	R	uint32	-	2	AQT base module firmware version hash number	
0x0054	R	uint16	-	1	HMP110 firmware version major number	
0x0055	R	uint16	-	1	HMP110 firmware version minor number	
0x0056	R	uint16	-	1	HMP110 firmware version revision number	
0x0057	RW	uint16	y	1	System time year	• Read = System time is read into cache when 0x0057 is read. Other registers return values from cache • Write = Register value is written to cache, and system time is updated when 0x005C is written
0x0058	RW	uint16	mo	1	System time month	
0x0059	RW	uint16	d	1	System time day	
0x005A	RW	uint16	h	1	System time hours	
0x005B	RW	uint16	min	1	System time minutes	
0x005C	RW	uint16	s	1	System time seconds	
0x005D–0x0064	-	-	-	-	Not in use	
0x0065	RW	uint16	-	1	Select the ppb to µg/m^3 conversion type for registers 0x0066–0x006B • 0 = EU standard (temperature t = +20 °C / +68 °F, pressure p = 1013.25 hPa) (default) • 1 = International standard (WHO, temperature t = +25 °C / +77 °F, pressure p = 1013.25 hPa) • 2 = Temperature based dynamic (ambient temperature from HMP110, pressure p = 1013.25 hPa) • 3 = Fully dynamic (ambient temperature from HMP110, pressure from an integrated pressure sensor)	
0x0066	R	int16	0.1 µg/m^3	1	NO_2 concentration calculated from ppb value with linear correction (register 0x0000) Calculated by means of selected conversion type. Calculation method is selected with register 0x0065	

Address	RW	Type	Unit	Register count	Description
0x0067	R	int16	0.1 µg/m^3	1	SO_2 concentration calculated from ppb value with linear correction (register 0x0001) Calculated by means of selected conversion type. Calculation method is selected with register 0x0065
0x0068	R	int16	1 µg/m^3	1	CO concentration calculated from ppb value with linear correction (register 0x0002) Calculated by means of selected conversion type. Calculation method is selected with register 0x0065
0x0069	R	int16	0.1 µg/m^3	1	H_2S concentration calculated from ppb value with linear correction (register 0x0004) Calculated by means of selected conversion type. Calculation method is selected with register 0x0065
0x006A	R	int16	0.1 µg/m^3	1	O_3 concentration calculated from ppb value with linear correction (register 0x0005) Calculated by means of selected conversion type. Calculation method is selected with register 0x0065
0x006B	R	int16	0.1 µg/m^3	1	NO concentration calculated from ppb value with linear correction (register 0x0006) Calculated by means of selected conversion type. Calculation method is selected with register 0x0065
0x006C	R	int16	ppb	1	NO_2 concentration in parts per billion (ppb), without linear correction • For a value with linear correction, use register 0x0000
0x006D	R	int16	ppb	1	SO_2 concentration in parts per billion (ppb), without linear correction • For a value with linear correction, use register 0x0001
0x006E	R	int16	ppb	1	CO concentration in parts per billion (ppb), without linear correction • For a value with linear correction, use register 0x0002
0x006F	-	-	-	-	Not in use
0x0070	R	int16	ppb	1	H_2S concentration in parts per billion (ppb), without linear correction • For a value with linear correction, use register 0x0004
0x0071	R	int16	ppb	1	O_3 concentration in parts per billion (ppb), without linear correction • For a value with linear correction, use register 0x0005

Address	RW	Type	Unit	Register count	Description
0x0072	R	int16	ppb	1	NO concentration in parts per billion (ppb), without linear correction • For a value with linear correction, use register 0x0006
0x0073	R	int16	0.1 µg/m^3	1	Particulate matter $PM_{2.5}$, without linear correction [1)] • For a value with linear correction, use register 0x0008
0x0074	R	int16	0.1 µg/m^3	1	Particulate matter PM_{10}, without linear correction [1)] • For a value with linear correction, use register 0x0009
0x0075	R	int16	0.1 µg/m^3	1	Particulate matter PM_1, without linear correction [1)] • For a value without linear correction, use register 0x0037
0x0076	R	int16	-	1	LPC data state • 0 = LPC data not ready • 1 = LPC data ready
0x0077–0x007A	-	-	-	-	Not in use
0x007B	R	int16	-	1	LPC fog present flag Combines all PM values from registers 0x007C–0x007E • 0 = Fog not present • 1 = Fog present, particle measurement values are invalid
0x007C	R	int16	-	1	LPC fog flag for PM_1 value • 0 = Fog does not affect PM_1 measurement • 1 = PM_1 measurement may be invalid
0x007D	R	int16	-	1	LPC fog flag for $PM_{2.5}$ value • 0 = Fog does not affect $PM_{2.5}$ measurement • 1 = $PM_{2.5}$ measurement may be invalid
0x007E	R	int16	-	1	LPC fog flag for PM_{10} value • 0 = Fog does not affect PM_{10} measurement • 1 = PM_{10} measurement may be invalid
0x007F	RW	int16	min	1	LPC measurement interval in minutes (2–255) Default: 10 minutes
0x0080–0x0085	-	-	-	-	Not in use
0x0086	RW	int16	%	1	CO linear correction gain value (1–255, corresponding gain values 0.01–2.55)
0x0087	RW	int16	%	1	O_3 linear correction gain value (1–255, corresponding gain values 0.01–2.55)
0x0088	RW	int16	%	1	NO_2 linear correction gain value (1–255, corresponding gain values 0.01–2.55)

Address	RW	Type	Unit	Register count	Description
0x0089	RW	int16	%	1	SO_2 linear correction gain value (1–255, corresponding gain values 0.01–2.55)
0x008A	RW	int16	%	1	NO linear correction gain value (1–255, corresponding gain values 0.01–2.55)
0x008B	RW	int16	%	1	H_2S linear correction gain value (1–255, corresponding gain values 0.01–2.55)
0x008C	RW	int16	ppb	1	CO linear correction offset –10000 ... 10000
0x008D	RW	int16	ppb	1	O_3 linear correction offset –10000 ... 10000
0x008E	RW	int16	ppb	1	NO_2 linear correction offset –10000 ... 10000
0x008F	RW	int16	ppb	1	SO_2 linear correction offset –10000 ... 10000
0x0090	RW	int16	ppb	1	NO linear correction offset –10000 ... 10000
0x0091	RW	int16	ppb	1	H_2S linear correction offset –10000 ... 10000
0x0092	RW	int16	‰	1	$PM_{2.5}$ linear correction gain value (per mille) Valid range: 1 ... 10000 (0.001 ... 10.000)
0x0093	RW	int16	‰	1	PM_{10} linear correction gain value (per mille) Valid range: 1 ... 10000 (0.001 ... 10.000)
0x0094	RW	int16	0.1 µg/m^3	1	$PM_{2.5}$ linear correction offset Valid range: -10000 ... 10000 (-1000.0 ... 1000.0)
0x0095	RW	int16	0.1 µg/m^3	1	PM_{10} linear correction offset Valid range: -10000 ... 10000 (-1000.0 ... 1000.0)
0x0096	RW	int16	‰	1	PM_1 linear correction gain value (per mille) Valid range: 1 ... 10000 (0.001 ... 10.000)
0x0097	RW	int16	0.1 µg/m^3	1	PM_1 linear correction offset Valid range: -10000 ... 10000 (-1000.0 ... 1000.0)
0x0098	R	uint32	s	2	Device uptime in seconds from last power-up
0x009A–0x00B3	-	-	-	-	Not in use
0x00B4	R	char[8]	-	4	AQT product device serial number (8-character string)
0x00B8	R	char[8]	-	4	HMP110 serial number (8-character string)

Address	RW	Type	Unit	Register count	Description
0x00BC	R	char[8]	-	4	LPC serial number (8-character string)
0x00C0–0x00C2	-	-	-	-	Not in use
0x00C3	R	int16	-	1	Circuit board serial number, part 1
0x00C4	R	int16	-	1	Circuit board serial number, part 2
0x00C5	R	int16	-	1	Circuit board serial number, part 3
0x00C6–0x00EF	-	-	-	-	Not in use
0x00F0	R	uint32	µV /mV	2	LPC photodiode backlight level in microvolts
0x00F2–0x00F3	-	-	-	-	Not in use
0x00F4	R	uint16	-	1	LPC firmware version major number
0x00F5	R	uint16	-	1	LPC firmware version minor number
0x00F6	R	uint16	-	1	LPC firmware version build number
0x00F7	R	uint32	-	2	LPC firmware version hash number
0x00F9	-	-	-	-	Not in use
0x00FA	W	uint16	-	1	Reset device • 1 = Reset device • Other values = no effect
0x00FB–0x00FD	-	-	-	-	Not in use
0x00FE	W	uint16	-	1	Write configuration to the non-volatile memory • 1 = Write configuration • Other values = no effect

1) *When PM concentrations reach the upper measurement limit of the laser particle counter (LPC) due to some exceptional event, the PM readings may temporarily be negative and the LPC reboots itself. For example, the unloading of sand in a nearby construction site and the resulting sand dust may cause the particle concentration to be temporarily drastically different from normal. Because of this the device initially shows very high PM readings and then negative readings. The device typically recovers from these exceptional events back to normal operation.*

Table 49 Modbus register addresses 0x1000–0x100D

Address	RW	Type	Unit	Register count	Description
0x1000	R	uint32	1/l	2	LPC particle bin number concentration, bin 1 Differential non-cumulative particle number from 0.27 µm to 0.6 µm (particles per liter)
0x1002	R	uint32	1/l	2	LPC particle bin number concentration, bin 2 Differential non-cumulative particle number from 0.6 µm to 1.0 µm (particles per liter)

Address	RW	Type	Unit	Register count	Description
0x1004	R	uint32	1/l	2	LPC particle bin number concentration, bin 3 Differential non-cumulative particle number from 1.0 µm to 1.6 µm (particles per liter)
0x1006	R	uint32	1/l	2	LPC particle bin number concentration, bin 4 Differential non-cumulative particle number from 1.6 µm to 2.5 µm (particles per liter)
0x1008	R	uint32	1/l	2	LPC particle bin number concentration, bin 5 Differential non-cumulative particle number from 2.5 µm to 4.0 µm (particles per liter)
0x100A	R	uint32	1/l	2	LPC particle bin number concentration, bin 6 Differential non-cumulative particle number from 4.0 µm to 6.3 µm (particles per liter)
0x100C	R	uint32	1/l	2	LPC particle bin number concentration, bin 7 Differential non-cumulative particle number from 6.3 µm to 10 µm (particles per liter)
0x100E	R	int16	0.1 µg/m^3	1	Particulate matter PM_1 1-hour average, with linear correction If the PM_1 value is invalid, this register returns −32768. Time is synchronized with calendar time.
0x100F	R	int16	0.1 µg/m^3	1	Particulate matter $PM_{2.5}$ 1-hour average, with linear correction If the $PM_{2.5}$ value is invalid, this register returns −32768. Time is synchronized with calendar time.
0x1010	R	int16	0.1 µg/m^3	1	Particulate matter PM_{10} 1-hour average, with linear correction If the PM_{10} value is invalid, this register returns −32768. Time is synchronized with calendar time.
0x1011	R	int16	0.1 µg/m^3	1	Particulate matter PM_1 1-hour average, without linear correction Time is synchronized with calendar time.
0x1012	R	int16	0.1 µg/m^3	1	Particulate matter $PM_{2.5}$ 1-hour average, without linear correction Time is synchronized with calendar time.
0x1013	R	int16	0.1 µg/m^3	1	Particulate matter PM_{10} 1-hour average, without linear correction Time is synchronized with calendar time.

Address	RW	Type	Unit	Register count	Description
0x1014	R	int16	-	1	Fog flag for PM_1 1-hour average value • 0 = Fog does not affect PM_1 1-hour average value • 1 = Fog affects PM_1 1-hour average value
0x1015	R	int16	-	1	Fog flag for $PM_{2.5}$ 1-hour average value • 0 = Fog does not affect PM_1 1-hour average value • 1 = Fog affects PM_1 1-hour average value
0x1016	R	int16	-	1	Fog flag for PM_{10} 1-hour average value • 0 = Fog does not affect PM_1 1-hour average value • 1 = Fog affects PM_1 1-hour average value

7.6 Setting system time over Modbus interface

You can read and write the system time over the Modbus interface.

The Modbus register addresses for this operation are:
- 0x0057
- 0x0058
- 0x0059
- 0x005A
- 0x005B
- 0x005C

1. To read system time, read register 0x0057.

   This reads the current system time to a cache and prevents the value from changing when other system time registers are accessed.
2. To get month, day of month, hours, minutes, and seconds, read registers 0x0058–0x005C.

   The reading order of these registers is free.
3. To write the system time, set the year, month, day of month, hours, and minutes with registers 0x0057–0x005B.

   This writes the date to the cache.

4. To finish updating the system time, write seconds to register 0x005C.

   This updates the system time immediately by applying all the recent values from the cache.

More information

- Setting system time (page 25)

8. Troubleshooting

If the product does not work as it should, check all cables and connectors.

Check the maintenance needs.

If the failure persists, contact Vaisala technical support.

Table 50 Troubleshooting

Problem	Probable cause	Remedy
Measurement failure or irrelevant data values.	The settings are incorrect.	Contact Vaisala technical support.
	There is a hardware failure.	
No response to any commands.	Wiring is wrong.	Check the wiring.
	Operational power is not connected.	Check the operating voltage.
	Baud rate or other serial property of the device is different from the host.	Connect the serial cable and check the serial port settings of the device with a terminal program.
No response to Modbus query.	RS-485 wiring is wrong.	Do the following: 1. Configure human-readable ASCII CSV report, see Configuring ASCII CSV data sending (page 21). 2. Verify that ASCII CSV report is sent every 60 s to RS-485 port. If data is not coming through, change the wiring of pins 5 and 6. See M12 pinout and wiring (page 65). 3. Configure original Modbus mode (Modbus RTU or Modbus ASCII) back to the device, see Configuring Modbus interface (page 20).

8.1 Checking device health information

The Modbus register address for this operation is 0x001F.

1. Establish a connection to the device.

2. To check the device health with CLI, type:

```
$ show health
```

   Example response.

```
show: health=99%
$
```

3. If the device health (service indicator) of your AQT560 is lower than 30 % (Modbus register address 0x001F), replace the base module.

   For instructions, see *AQT560 Maintenance Guide (M212983EN)*.

8.2 Writing a problem report

When troubleshooting the product, write a problem report including:

- What failed (what worked / did not work)?
- Where did it fail (location and environment)?
- When did it fail (date, immediately / after a while / periodically / randomly)?
- How many failed (only one defect / other same or similar defects / several failures in one unit)?
- What was done when the failure was noticed?
- What was connected to the product and to which connectors?
- Input power source type, voltage, and list of other items (such as lighting, heaters, and motors) that were connected to the same power output.
- Are all parts connected and grounded properly? Take a photo to help the troubleshooting.

This information is helpful if you need to contact Vaisala support.

AKRIBIS · www.akribis.info · contacto@akribis.info

Appendix A. M12 pinout and wiring

The M12 connector provides RS-232 and RS-485 serial interfaces.

The 8-pin M12 connector is located on the side of the air quality transmitter and provides power and data to the transmitter.

The following pinout is from the side of the device, not the cable.

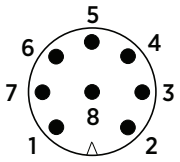


Figure 5 Pinout of M12 connector

Table 51 Pinout of M12 connector

M12 pin	Wire color	RS-232 (CLI)	RS-485 (Modbus, CSV ASCII)
1	White	Data GND	Data GND
2	Brown	RX (input)	–
3	Green	TX (output)	–
4	Yellow	No connection (floating)	No connection (floating)
5	Gray	–	A (D–)
6	Pink	–	B (D+)
7	Blue	Power GND	Power GND
8	Red	Power 10–25 V DC	Power 10–25 V DC

RS-232 serial interface

The RS-232 interface is used mainly for the maintenance connection.

RS-485 serial interface

The RS-485 interface is used mainly for the data connection.

RS-485 supports Modbus ASCII and Modbus RTU, and ASCII formatted text with comma separated values (CSV).

Appendix B. OSS licenses

This product contains open source software (OSS) components. Such OSS is governed by the terms and conditions of the applicable OSS licenses, and you are bound by the terms and conditions of such licenses in connection with your use and distribution of the OSS in this product.

You may request a copy of certain open source code as required by certain applicable OSS licenses by sending an email to opensourcesw@vaisala.com with the following subject line: *OSS source code request*.

List of installed packages and their respective versions and licenses:

Table 52 AQT560 OSS license

Library	Version	License
avr-libc	2.0.0	Modified BSD License / https://www.nongnu.org/avr-libc/LICENSE.txt

AKRIBIS · www.akribis.info · contacto@akribis.info

License text

The contents of avr-libc are licensed with a Modified BSD License.

All of this is supposed to be Free Software, Open Source, DFSG-free, GPL-compatible, and OK to use in both free and proprietary applications.

See the license information in the individual source files for details.

Additions and corrections to this file are welcome.

****
Portions of avr-libc are Copyright (c) 1999-2010
Keith Gudger,
Bjoern Haase,
Steinar Haugen,
Peter Jansen,
Reinhard Jessich,
Magnus Johansson,
Artur Lipowski,
Marek Michalkiewicz,
Colin O'Flynn,
Bob Paddock,
Reiner Patommel,
Michael Rickman,
Theodore A. Roth,
Juergen Schilling,
Philip Soeberg,
Anatoly Sokolov,
Nils Kristian Strom,
Michael Stumpf,
Stefan Swanepoel,
Eric B. Weddington,
Joerg Wunsch,
Dmitry Xmelkov,
The Regents of the University of California.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holders nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

****

Warranty

For standard warranty terms and conditions, see www.vaisala.com/warranty.

Please observe that any such warranty may not be valid in case of damage due to normal wear and tear, exceptional operating conditions, negligent handling or installation, or unauthorized modifications. Please see the applicable supply contract or Conditions of Sale for details of the warranty for each product.

Technical support



Contact Vaisala technical support at helpdesk@vaisala.com. Provide at least the following supporting information as applicable:

- Product name, model, and serial number
- Software/Firmware version
- Name and location of the installation site
- Name and contact information of a technical person who can provide further information on the problem

For more information, see www.vaisala.com/support.

Recycling



Recycle all applicable material according to local regulations.

AKRIBIS · www.akribis.info · contacto@akribis.info

VAISALA

www.vaisala.com

AKRIBIS · www.akribis.info · contacto@akribis.info